STOCHASTIC CAPABILITY MODELS FOR DEGRADING SATELLITE CONSTELLATIONS

THESIS

Cole W. Gulyas, Captain, USAF

AFIT/GOR/ENS/05-07

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT/GOR/ENS/05-07

STOCHASTIC CAPABILITY MODELS FOR DEGRADING SATELLITE CONSTELLATIONS

THESIS

Presented to the Faculty
Department of Operational Sciences
Graduate School of Engineering and Management
Air Force Institute of Technology
Air University
Air Education and Training Command
in Partial Fulfillment of the Requirements for the
Degree of Master of Science in Operations Research

Cole W. Gulyas, B.S.
Captain, USAF

March 2005

AFIT/GOR/ENS/05-07

STOCHASTIC CAPABILITY MODELS FOR DEGRADING SATELLITE CONSTELLATIONS

Cole W. Gulyas, B.S.

Captain, USAF

Approved:

Dr. Jeffrey P. Kharoufeh Thesis Advisor	Date
Maj Stephen P. Chambal Ph.D. Committee Member	Date

AFIT/GOR/ENS/05-07

Abstract

This thesis proposes and analyzes a new measure of functional capability for satellite constellations that incorporates the instantaneous availability and mission effectiveness of individual satellites. The capability measure yields a continuous score between zero and one accounting for the degree to which the constellation meets operational requirements. The measure is computed from an average of satellite capabilities, composed of the product of the satellite's instantaneous availability and value score. Instantaneous availability is acquired by modeling the satellite degradation status as either a time-homogenous, continuous-time Markov chain (CTMC) if it possesses functions with exponential lifetime distributions, or as a time-homogenous, semi-Markov process (SMP) if the function lifetime distributions are not exponential. The satellite value score represents the individual satellite's contribution to the overall constellation mission and is obtained using multi-attribute value theory. For illustrative purposes, analytical results were compared with those obtained via the Monte Carlo method and were found to be indistinguishable with substantially less computational effort.

Acknowledgements

I would like to take this opportunity to thank a few people who supported me through times of course completion and the writing of this thesis. First, I thank my advisor Dr. Jeffrey Kharoufeh for his encouragement, enthusiasm, and direction throughout the writing of this thesis. With his comprehensive knowledge and high standards, he inspired me to produce the highest quality work I was capable of. Second, I thank my reader Maj Stephen Chambal for reviewing the thesis document and offering suggestions for improvement. Third, I thank Mr. Justin Comstock for his insight into the problem this thesis addresses. The many discussions we had were helpful in understanding the problem. Fourth, I thank my fellow students for their camaraderie during our time here at the Institute. Finally, I thank my wife and sons for the sacrifices they made during my tour here at AFIT. I can only hope that someday they will benefit from the experience I have gained while completing this work. Thank you.

Cole W. Gulyas

Table of Contents

Page

List of Figures

List of Tables

STOCHASTIC CAPABILITY MODELS
FOR DEGRADING SATELLITE CONSTELLATIONS

1. Introduction

1.1 Background

The unique vantage point of space offers numerous advantages to military forces operating in a global arena. Missions employed by the United States military in the present, and in the future, will rely heavily on information provided by space based systems. A minimum level of capability is required to support these missions, and the efficient management of maintaining this capability level into the future depends on an accurate capability assessment via performance and availability measures. Without an idea of how well these systems meet the requirements of the missions that rely on their services, the potential exists for excessive spending to ensure systems are capable, or systems may not be adequately capable to support military operations. Justification for funding new systems and maintaining current systems may occur due to the gap that exists between current capabilities and the requirements of the user.

Missions performed by U.S. military satellite constellations are summarized in the Joint Doctrine for Space Operations [5] published in August 2002. These satellite constellations perform many essential missions pertaining to intelligence, surveillance, and reconnaissance (ISR); integrated tactical warning and attack assessment; environmental monitoring; communications; and position, velocity, time, and navigation. The increased importance of these missions to the successful completion of military goals and the increased likelihood of attacks on space assets which support these missions has generated a need for an accurate measurement of a satel-

lite constellation's functional capability, a measure of the system's ability to perform its intended mission.

Deriving a functional capability methodology for satellite constellations is of interest to any company or agency that provides direct support to satellite constellations. Capability-based constellation replenishment models require an accurate assessment of a constellation's capability. This functional capability methodology may also help redefine maintenance and operation policies, increasing the life expectancy of the constellation by perhaps operating its satellites in a more efficient manner relative to the satellite's current mission and condition. Also, any sudden change in availability of the constellation may indicate if an attack on space assets has occurred and if so, when the attack occurred and to what degree the attack had on the system.

The incorporation of new technology and the sustainment of current capabilities involving space systems is paramount to the success of future military operations. Joint Vision 2020 [6] is a collaborative strategy composed by the Joint Chiefs of Staff that describes the necessary transformations of the joint military forces current capabilities for successful future military and humanitarian operations. The document states two important philosophies relative to military operations in space. First, information superiority, "the capability to collect, process, and disseminate an uninterrupted flow of information while exploiting or denying an adversary's ability to do the same [6:8]," is recognized as a necessary component in successful military operations. The publication emphasizes the adoption of a new doctrine dependent on the availability of timely and accurate information- information assumed to be readily available in a real time, error-free delivery. Second, in order to be successful against enemies of the future, military capabilities will have to be adequately dynamic to keep the enemy from being able to adapt and counteract our capabilities. The vision statement suggests that one of the best ways to keep military capabilities

from entering a static state is through "...the steady infusion of new technology and modernization and replacement of equipment" [6:3].

1.2 Problem Definition and Methodology

The United States Air Force has a requirement to measure satellite performance based on the capability of a satellite constellation, rather than measures derived from expected lifetime estimates [9]. This thesis will propose a model which describes satellite functional capability on a continuous scale and can assess the partial capability of a constellation assuming certain conditions.

As with all environments in which the military operates, space offers unique advantages and disadvantages that affect system functional capability. The obvious advantages to military systems include global *access*, the ability to have a line of sight to any point on the globe, and *persistence*, the ability of space assets to stay on orbit for long durations of time. There are also many disadvantages to operating in space that can have an effect on functional capability of these systems. First, space is a hostile environment which can cause system failures, disruptions, and degradation in performance, all of which directly affect the system's state of availability. Second, the location of space assets limits convenient access to these systems, resulting in high costs associated with fielding and maintaining them. Also, the amount of expendable resources these systems are designed to store is restricted due primarily to deployment restrictions associated with weight and the extent to which maintenance can be performed to replenish degraded systems.

Historically, metrics based solely on lifetime estimates have been used in place of capability-based measures for satellites, which were then pooled into a measure representing the constellation lifetime. Mean Mission Duration (MMD), the expected lifetime of the space asset, was used to estimate how long the system would be available. This measure, however, does not inherently incorporate how well the

system performs its mission. In most cases, MMD underestimates the lifetime of the system and lacks the ability to accurately predict the capability of the system at a future point in time [9]. Satellite systems are over-engineered to meet guaranteed satellite lifetime requirements. *Derating* is a method used to increase the reliability of a component by operating it at a significantly lower stress level than it was originally designed. Higher-rated components are used in place of normal rated components to increase the reliability and lifetime of the component. Over-engineering leads to systems that meet their required MMD specification with a high probability, making MMD a poor measure upon which to base an estimate of the system's true lifetime [9].

Most current methods of assessing constellation capability use either decision analysis techniques or reliability measures using steady-state system availability as the basis of their methodology. Multi-attribute value theory, a subset of decision analysis, may provide a methodology of scoring a system by assigning weights to certain attributes of the system directly correlated to a defined value, in the case of functional capability, a value representing the system performing its mission. The shift from alternative based thinking to value focused thinking provides an approach to assessing the value of a satellite to the overall constellation mission and may help in generating the overall measure of functional capability. The process involves first determining the appropriate attributes, usually through interviews with subject matter experts, assigning a score related to order of priority of these attributes, then determining a weight of importance for each attribute. After attributes are determined with their corresponding values and weights, a function is composed which yields a value between 0 and 1 that indicates how well the system is able to perform its mission. This approach yields an estimate of functional capability and is strictly confined to a particular constellation, since the attribute selection and scoring and weighting are dependent on the nature of a constellation mission.

The availability of a system is defined in terms of its reliability and maintainability, all of which are probability functions of time. *Reliability* is the probability that a system does not fail before a specific time; it is the complement to the probability that a system fails before a specific time, also known as the cumulative probability distribution function. *Maintainability* is the probability that a system's reliability can be restored to some level through maintenance or repair actions. *Instantaneous availability* is the probability a system is available to perform its designed mission at some point in time. *Steady-state availability* is the proportion of up time to the total time of operation over a very long period of time. If a system's steady-state availability exists, the system's instantaneous availability will eventually converge to its steady-state availability.

System effectiveness is a measure which gives the probability that a system will perform its intended mission under its designed operating conditions. Ebeling [14], suggests that system effectiveness is comprised of the system's operational readiness, the probability of the system working at the start of its life; the system's mission availability, the proportion of lifetime the system is performing its mission; and the system's design adequacy, the probability the system will perform as it was designed. In terms of satellite effectiveness in a constellation, the operational readiness is described as the probability the satellite is on station and working, surviving the launch, deployment, and initialization processes; mission availability is described as the proportion of the satellite's lifetime in the constellation performing its mission; and the satellite's design adequacy is described as the probability the satellite does what it was designed to do. If operational readiness, mission availability, and design adequacy are independent probabilities, one possible definition of satellite effectiveness might be the product of its respective probabilities of operational readiness, mission availability, and design adequacy.

The effectiveness measure results in a number between 0 and 1 in a multiplicative manner, which describes a level of effectiveness for the constellation with

no associated units. The system effectiveness measure is based on the engineering side of system development, where design adequacy is a percentage based on specifications that are more focused on design versus operational adequacy. Ebeling [14] suggests that one might obtain the design adequacy measure by finding the proportion of time the system adequately performs its intended job. However, a subject matter expert would need to be involved to make the distinction between what criteria to use in measuring adequacy. Decision analysis techniques could then be used to assess how well the system performed its job.

Efforts have been made to create a meaningful estimate of constellation functional capability which contain elements of reliability techniques combined with decision analysis techniques [9]. A score is formed by taking the product of three terms: the product of the reliability associated with the random and wear-out phases of the satellite's lifetime, which accounts for the probability of the satellite being on station and currently working at a specific time; the duty cycle ratio, the ratio of estimated mean duty cycle over the beginning-of-life duty cycle, which addresses the percent of time the satellite is performing its mission; and the band capability, which is a formulation of a subject matter expert's assessment of the satellite's payload capability.

A satellite's duty cycle is the proportion of time the satellite is performing its mission. When a satellite is in orbit, there may be cycles of time in which there are no tasks to be performed by the satellite. Changing the mode of the satellite to a standby status during periods of inactivity extends the overall lifetime of the satellite. Duty cycle is directly related to the reliability of the satellite over time; the higher the duty cycle, the faster the functions of the satellite will wear out.

The composite score was designed in the hopes of capturing the constellation's satellite equivalence, an estimate of the equivalence of how many fully capable satellites are in the constellation. It generates a number which represents the number of satellite equivalents in the constellation in an additive manner where equivalence is

relative to the capability level of the newest satellite. The number is then compared to a minimum number of satellites required to accomplish a mission. Theoretically, one could define intervals of satellite numbers which describe states of the constellation's capability to include fully capable, partially degraded, and fully degraded states.

The reliability measures for the random and wear-out phases of the satellites are formed by statistical models based on historical observation data from similar satellites. These models are open to errors associated with small, dependent samples from systems which may not possess reliability characteristics of the systems that are being modeled.

Basing the composite score on a theoretical number of satellites in a constellation would be acceptable if the number of satellites is a representative estimate for the constellation's capability. This may or may not be a good measure for some constellations due to the nature of their missions as well as the nature of their configurations. For example, in a GPS constellation, numeracy has a definite correlation to area coverage due to line-of-sight restrictions in the transfer of data between ground and space. Considering a measure which represents a satellite equivalency makes sense for such a constellation since the number of satellites holds direct relation with the constellation's capability. However, there may be instances when this measure may not fully explain the satellite capability. For example, if there exist satellites within the constellation that depend on other relay satellites for communication with ground stations, these relay satellites hold more value in terms of the mission, and a score based on the number of satellite equivalents does not differentiate between non-identical satellites.

The goal of this thesis is to present a methodology that combines the capability assessment of decision analysis techniques with reliability theory to develop a measure that meets the requirements of the United States Air Force. This measure will reflect the ability of a constellation to fulfill mission requirements and it will

encompass the probability associated with the constellation being available to perform its mission. The methodology will assess the constellation at the satellite level, where methods for deriving its *instantaneous availability* will be the main focus. Using instantaneous availability gives a more representative measure of availability for newer constellations that have been on orbit for a relatively short amount of time. The decision analysis methodology will be applied to the satellite's operational contribution to the constellation, focusing on the extent to which the satellite accomplishes constellation mission requirements, rather than its ability to meet design specifications.

1.3 Thesis Outline

The next chapter will provide an overview of the literature which contributes to the development of functional capability measures, including basic concepts of reliability and multi-attribute value theory followed by a summary of methods that attempt to optimize constellation replenishment policies and compute constellation performance measures. Chapters 3 and 4 will discuss the methodology behind the new constellation functional capability measure. Chapter 5 will apply the methodology to navigation, communication, and meteorological satellite constellation examples to illustrate the methodology and to compare analytical results with simulation results, demonstrating advantages of the analytical methods. Finally, Chapter 6 will summarize the results and offer suggestions for further research.

2. Review of the Literature

This chapter will discuss the literature that has a direct contribution to the understanding and development of a satellite constellation functional capability measure. The first section will present a timeline of general developments in the field of reliability pertaining to the characterization of system lifetimes. The second section will focus on the main developments of availability. The third section will discuss some of the current constellation replenishment policies in the literature. The fourth section will end the chapter with a discussion on the methodologies related to obtaining a satellite capability measure either by applying the theory of reliability or multi-attribute utility theory.

2.1 Reliability

Reliability theory is the application of probability and statistics in determining characteristics of a system's failures over time. The development of this theory started during our nation's transition from an agricultural to an industrial based economy. A need for machines to be maintained in an efficient manner was key to a profitable manufacturing process. One branch of the development of reliability focuses on the lifetime of systems, while another focuses on characterizing an optimal policy of repairing and replacing these machines. The paper by Barlow [2] gives a history of the main developments in reliability theory.

Reliability is a probability function of time that indicates the likelihood of a system not failing before a certain time. This function is sometimes also called the *survivor* function, as it characterizes the probability the system will survive up to a point in time. The complement of the reliability function is the lifetime cumulative distribution function (c.d.f.).

The early developments of reliability started in the 1930s and were based on finding reliability functions that described the fatigue life of materials. The jus-

tification of modeling incoming telephone calls as Poisson arrivals by Erlang and Palm laid the foundation for arguments supporting the use of exponential functions to describe system lifetimes [2]. In the 1950s, Epstein and Sobel [15] initiated the assumption that the exponential distribution was most applicable to model system reliability, which was later supported by Davis in 1952 when he published a paper characterizing the fit of several reliability functions to actual failure data [12]. The exponential function was popular not only because of the research supporting it, but also due to the simple results obtained with its use. In 1939, Weibull characterized a different function which described the breaking strengths of materials [40]. This reliability function was first adopted due to its simplicity and was later given more attention by papers from Kao [27] and from Zelen and Dannemiller [43] to be more robust than the exponential function in certain applications.

2.2 Availability

Availability is the probability a system is available to perform its intended function at some time t, given that the system may have failed and received a repair in the past. It is a measure that is closely tied to the concept of system reliability by encompassing the maintainability aspect of a system by by accounting for the increases in reliability caused by repairs or replacements. The availability measure will always be greater than or equal to the reliability measure; if a system cannot be repaired or replaced, its availability will be equal to its reliability. The branch of reliability theory characterizing replacement policies started in the 1940s with the research of Lotka [31] and Campbell [4].

The modeling of a system's availability is usually approached using a combinatorial model, a state-space model, or a combination of the two. Combinatorial models include reliability block diagrams, reliability graphs, and fault trees. These methods were developed in the 1970s, were the first ideas of fault tree analysis were presented by Fussell and Vesely [20] describing minimum cut sets for fault trees. The idea

of combinatorial models stems from the assumption that the failure of components of a system are independent of one another. By arranging these components into combinations of serial and parallel configurations, and assigning reliability measures to each component, the system's availability can be determined. By enumerating the possible events that can cause a system failure, the probability of a system failure can be computed using the basic laws of probability.

State-space models allow the modeling of interactions between component failures. A typical model consists of describing the system in terms of states that represent the condition in which the system may be found. The system's degradation can then be modeled as a stochastic process. The probability of the system being in a state at a given time may be calculated based on information on the rates of transitions, provided the stochastic model possesses certain characteristics. One of the first papers regarding stochastic modeling of system reliability was written by Weiss in 1956, where he used semi-Markov processes to characterize system replacement policies [41]. Two examples of modeling the availability of systems using state-space models are given in papers by Fricks [19] and Ibe [26]. Fricks discusses the availability analysis of a multi-computer system while Ibe describes the availability modeling of a management system which automates the operation of a utility company.

2.3 Satellite Constellation Replenishment Models

There are many papers written on the subject of satellite constellation replenishment. These replenishment models are based on optimizing a certain aspect of the constellation, whether it be minimizing cost or maximizing capability. These models depend on a performance measure on which to base the optimization model.

For example, in a paper by Collopy [8], a replacement model is developed for a satellite constellation where the problem of determining the number of optimal spares

is addressed. The policy is based on keeping the satellite constellation in a capable state, defined by a minimum number of operating satellites. The constellation has $n = m + i$ satellites where m is equal to the minimum number of required satellites, and i is equal to the number of on-orbit spares. The methodology starts by looking at the event in time in which a satellite in the constellation fails. At this point, a spare replaces the failed satellite and the number of satellites in the constellation is reduced to $n = m + i - 1$. The paper addresses the probability of running out of spares and falling below the minimum number of operating satellites, n, before a satellite replacement can be launched to replenish the constellation. This interval is from time of failure to replenishment of the constellation by a non-on-orbit satellite is referred to as *Replacement Launch*. The number of failures in that interval has a Poisson distribution with parameter $\lambda = n \cdot p$, where p is the ratio of *Replacement Launch* over the MTTF of any given satellite, and n is the number of satellites left in the constellation; satellites are considered to be independent and identical.

The probability of running out of spare satellites before a satellite can be launched to replenish the constellation is equivalent to a system failure and is given as

$$P(i) = e^{-\lambda} \cdot \frac{\lambda^i}{i!},$$

which is the probability of i failures, and does not account for any probability associated with more than i failures.

The paper then defines the term *Cycle* as being equal to satellite reliability, $R(t)$ divided by the number of satellites in orbit prior to the failure, added with the time interval *Replacement Launch*; *Cycle* is the time interval in which the probability of system failure is based on, where

$$Cycle = \frac{R(t)}{m + i} + Replacement\ Launch.$$

The system MTBF is given as the *Cycle* divided by the probability of a system failure occurring, where

$$\text{MTTF}_s = \frac{Cycle}{P(i)} = Cycle \cdot e^{\lambda} \cdot \frac{i!}{\lambda^i}.$$

Collopy [8] uses optimization techniques to minimize the number of spares maintained in the constellation through an objective function which captures the cost of the system based on cost of individual satellites and cost of a system failure. The model assumes the satellite constellation is "capable" based on a minimum number of operating satellites and does not address partially operating satellites. Failed satellites are replaced by spares which are either on orbit or are waiting to be launched. The satellite lifetimes are modeled as exponential random variables, and all satellites are independent and identical.

An example of applying replenishment methodology to satellite constellations is given by Feuchter [17]. He discusses the conditions when a constellation should be maintained by replacing failed satellites or by implementing on-orbit repairs to degraded satellites. An optimal replacement policy attempts to prescribe when it is more economical to repair the satellite versus simply replacing it. Examples of satellites designed for on-orbit repair are given as a contrast to the types of satellites the Department of Defense typically uses. For example, the NASA program satellites such as the Hubble Space Telescope (HST), the Gamma Ray Observatory (GRO), the Advanced X-Ray Astrophysics Facility (AXAF), and the Space Infrared Telescope Facility (SIRTF) are large, costly satellites, whereas the Department of Defense typically uses cheaper, constellation-based satellites, which are designed under a replacement policy and are deployed in orbits that have limited access.

Although the Department of Defense does not plan on converting over to a repair policy in the near future, Feuchter's paper attempts to define the boundary between the two policies by considering existing constellations, culminating in the

Comprehensive On-orbit Maintenance Assessment (COMA) final report. “COMA focuses on constellation support, which consists of establishing a constellation and keeping it functional throughout its life” [17:499]. It relies on resource requirements generated by a simulation model, which consists of a function of constellation availability (differential lifecycle cost model). Satellite mass turns out to be a differentiator between expendable and repairable satellites and as such, is an important model input.

The satellite is broken down into six subsystems; the structure, attitude control system, communications, telemetry tracking and command, electrical power system, and sensor subsystems. The paper also makes the distinction between satellites as either being a sensor-type satellite or a communications satellite; sensor-type having a sensor based payload with a limited communications subsystem and communication having no sensors, with all communications payload. Tactics and strategies are related to parameters “that describe the size, shape, and spatial orientation of the satellite orbit” [17:501]. Strategies consider schedule maintenance, reactive repair policies, and policies involving repairs before a failure is anticipated. Tactics include ”expendable and reusable transfer vehicles” and ”one or two orbital platforms per scenario to serve as warehouses and transportation nodes” [17:501].

The truncation of a satellite’s lifetime is stated as being relatively predictable, since it is based on deterministic factors such as “mechanical wearout or the exhaustion of consumables” [17:502]. Also, support cost was determined to be statistically significant with the following four constellation parameters and five satellite parameters: number of satellites in constellation, constellation maintenance time, cost of transportation to constellation, location, constellation transportation efficiency, satellite mass, modularization mass penalty, reliability, truncation lifetime, and measure of value of retrieving failed satellites and modules. They used fractional factorial designs to minimize run lengths.

An article by Hopp [25] discusses incorporating technological improvements in a replenishment model for a general system by basing it on the obsolescence rather than the deterioration of the equipment. The paper states that past replacement methods are based on equipment wear and assume that replacements are identical to the previous equipment. Previous papers attempted to incorporate the obsolescence of systems by assuming that improving technology and deteriorating equipment were deterministic, while [21] attempted to model the technology improvement as a stochastic process with deterministic equipment deterioration. Hopp takes the next step of modeling both stochastic technological improvement and deterioration.

A single technological improvement is assumed to occur in an unknown time, resulting in a recursive, non-time homogeneous optimal value function that is solved by restricting the function to an initial state and applying theory of forecast horizons. The function makes the decision of keeping or replacing the equipment. New technology-based replacements are assumed to be better than the legacy equipment. The paper goes on to show the effects of technological change on the replacement policy, finding that available technology fosters early replacement while the possibility of technology being available in the future fosters incentive to keep current technology longer. Methods are discussed regarding using the recursion function with time in reverse to calculate the expected cost of a sure event of technological improvement in the future, given the use of old technology in the present.

The paper continues with methods of computing the optimal replacement policy based on minimizing the net present cost based on the recursion formulas. They establish that a forecast horizon will exist if the optimal replacement policy is unique, and they define an efficient stopping criteria for the recursive algorithm that includes the chance of the function converging on a solution where both the *keep* and *replace* portions of the function are equal. The two stopping rules Bound Based and Bés and Lasserre's rule are compared, resulting in the preference to the Bound Based rule due to its computational efficiency and minimal data requirements.

2.4 Satellite Capability Measures

Many terms have been used to capture the aspects of a system not accounted for by its reliability. *System effectiveness* is a term used by Ebeling [14] to describe a system's probability of performing its intended mission, incorporating the system's performance and availability. *Performability* is a related measure to system effectiveness which is a composite measure of the performance and reliability of a system. A composite measure of availability and performability captures a system's effectiveness.

Meyer [32] gives a formal presentation of the ideas of performability, effectiveness, and capability. The author describes perfomability in terms of a probability measure mapping a set of events to a probability space. Performability is then a function mapping an accomplishment set, referring to measures of system accomplishment, to a probability measure. He defines a *capability function* as the mapping from a state trajectory from a stochastic model to a level of accomplishment.

There are many examples of papers that focus on measuring qualities of satellite constellations, whether they be termed measures of effectiveness, capability, or performance. In [34], Smith models the performability of a multiprocessor system using a Markov reward model (MRM), which incorporates rewards with the state-spaces of a stochastic process.

The thesis by Wilson [42] focuses on the performability of a geosynchronous weather satellite system, although he refers to the resultant measure as a measure of effectiveness (MOE). He uses decision analysis methods in conjunction with published sources along with subject matter expert opinions on what defines the quality of the system to score the constellation. A list of attributes are first defined which represent system parameters that have a direct impact on satellite weather systems. Then these parameters are evaluated in reference to area coverage of the earth. Wil-

son gives a hierarchical model of the weather system's mission which ties the broad goals of the mission to measurable performance parameters.

A thesis by Staats [35] developed a measure of a satellite's utility through multi-attribute utility theory. The reliability of the satellite is incorporated into the measure by introducing an attribute reflecting the SME's opinion of the satellite's expected remaining lifetime. Although the reliability is not explicitly computed, the resulting measure comes closer to a desired measure of satellite performance. The terms *value* and *utility* are often used interchangeably; however, this thesis addresses strictly the value of the satellite with respect to its contribution to the constellation mission.

2.5 Summary and Contributions

Constellation-specific methodologies incorporating maintenance aspects of satellites are either replenishment policies determining the level of spares to maintain, or policies deciding when to optimally replace or repair a satellite based on net present cost. If a replenishment policy is to be based on capability, a constellation capability measure will be required.

Multi-attribute utility theory has been the most common method of computing measures of satellite performance. One method reviewed attempted to incorporate reliability into the satellite value using utility theory, while another method ignored aspects of reliability. A transient model of satellite degradation was not found in the literature, nor was a measure of constellation capability based on the availability of satellite functions.

It is clear from the literature that constellation capability, a measure of the performability of the satellite constellation, should incorporate both aspects of its reliability and mission effectiveness. This research offers a stochastic model of the degradation process of a constellation's satellites, from which a measure of availabil-

ity is computed. Value scores of satellite capability are incorporated into the measure using value theory to yield a reliability-based performance measure of constellation capability. The next chapter will present a formal mathematical model that may be used to compute a new constellation functional capability measure.

3. Formal Mathematical Model

This chapter will give a description of the formal mathematical model for constellation capability. By modeling the status of each satellite in the constellation as a time-homogeneous, continuous-time Markov chain (CTMC), and by using decision analysis techniques to subjectively score each satellite, we derive a methodology that provides a continuous measure of capability for a constellation. Each satellite's instantaneous availability and value score are evaluated as inputs into the functional capability measure. The instantaneous availability is a function of time that measures the satellite's probability of being available at a future point in time. The value score is a measure between 0 and 1 representing the satellite's capability to perform its mission within the constellation based on a subject matter expert's opinion.

3.1 Constellation Functional Capability

A satellite constellation operates in the environment of space which imposes stresses on its satellites that may potentially result in a degradation of functional capability. It is impossible to completely determine how long a satellite constellation will remain in a functionally capable state because both the constellation and the environment in which it operates in are not completely determined; there exist random elements in the system and in the environment that make it difficult to characterize the behavior of the system over time. By probabilistically modeling a satellite in a constellation, and by making a few reasonable assumptions about the satellites in the constellation, we are able to develop a mathematically tractable model which gives the instantaneous availability and a measure of capability for the satellite constellation. Instantaneous availability is chosen because it is a more accurate measure of availability for systems which have not been in service for a long period of time.

This section describes the constellation functional capability measure $\Phi(t)$, composed of each satellite's instantaneous availability score and value score. The

constellation functional capability is the mathematical average of the functional capability scores for the satellites within the constellation. The functional capability measure for an arbitrary satellite, namely satellite k, is the product of its instantaneous availability at time t, denoted $A^{(k)}\left(t - t_0^{(k)}\right)$, and its value score, denoted $V^{(k)}(t)$. The value $A^{(k)}\left(t - t_0^{(k)}\right)$, which is based on the time of its initialization $t_0^{(k)}$, is a probability and as such, may assume a value between 0 and 1. The initialization time $t_0^{(k)}$ is the time at which the satellite's life in the constellation begins. The value score $V^{(k)}(t)$ is a score which describes the functional value of the satellite to the constellation mission and is derived through multi-attribute value theory. It is dependent on the time the constellation is evaluated and can also take on a value between 0 and 1. Given the satellite constellation consists of K satellites,

$$\Phi(t) = \frac{1}{K}\sum_{k=1}^{K} A^{(k)}\left(t - t_0^{(k)}\right) V^{(k)}(t). \tag{3.1}$$

This average of constellation functional capability is dependent on the time at which the "measurement" is taken, and on the set of criteria comprising the value score $V^{(k)}(t)$. The capability at time t, $\Phi(t)$, will be a measure between 0 and 1, where 0 corresponds to a constellation that has no functional capability and 1 corresponds to a constellation that has full functional capability. For example, if each satellite in the constellation is completely available at time t, and each satellite is also scored as having perfect value, then

$$A^{(k)}\left(t - t_0^{(k)}\right) = 1, \quad V^{(k)}(t) = 1, \quad k = 1, 2, \ldots, K$$

and

$$\Phi(t) = \frac{1}{K}\sum_{k=1}^{K} A^{(k)}\left(t - t_0^{(k)}\right) V^{(k)}(t) = \frac{1}{K}\sum_{k=1}^{K}(1)(1) = \frac{1}{K}\cdot K = 1.$$

If each satellite is unavailable,

$$A^{(k)}\left(t - t_0^{(k)}\right) = 0, \quad k = 1, 2, \ldots, K$$

and

$$\Phi(t) = \frac{1}{K}\sum_{k=1}^{K} 0 \cdot V^{(k)}(t) = 0.$$

The constellation functional capability measure $\Phi(t)$ captures both the mission contribution and the availability of the satellite, where the score takes into account aspects of each satellites reliability and fulfillment of operational requirements.

The following sections will focus on the satellite level of the constellation and will discuss the assumptions and methodology for obtaining the instantaneous availability and value score for the satellites when the degradation status of each satellite is modeled as a continuous-time Markov chain (CTMC). Since the literature is quite extensive in the area of assigning a value score to a satellite, the methodology for producing a value score is covered with less emphasis than the methodology for obtaining its instantaneous availability measure.

3.2 Instantaneous Availability

A model for instantaneous availability is given for a general satellite constellation consisting of K satellites, where the degradation state of a satellite can be modeled as a time-homogeneous, continuous-time Markov chain (CTMC); the failure and repair rate distributions of the satellite states are restricted to exponential distributions. Each satellite is considered independent of the others and has $M^{(k)}$ main functions that can be in only one of two states: operational and non-operational. A more general model is presented in Chapter 4 for which the function failure and repair distributions are not restricted to exponential distributions.

3.2.1 Stochastic Degradation Model

The following assumptions apply to the CTMC model for computing instantaneous satellite availability:

1. The constellation under consideration contains a finite number (K) of independent satellites, each with a set of finite functions.
2. The failure and repair rates of each function of the satellite are assumed to be known.
3. The distribution of failure and repair times is exponential.
4. Multiple satellite function failures or repairs cannot occur in any instant of time.
5. The initial state of the satellite constellation is assumed to be known; the states the satellite may be in are assumed to be definable and the probabilities of being in these states at time $t_0^{(k)}$, the beginning of the satellite's lifetime, are assumed to be known.
6. The states of availability of the satellite can be determined based on the availability of specific functions.

The models presented in this chapter and in Chapter 4 will not consider the possibility of on-orbit spares. A different methodology tailored specifically for handling spares and for addressing long-term constellation replenishment policies is provided, for example in [8], and may be more appropriate for assessing constellation availability when considering satellite replacement. The methodology presented in this and the following chapter measures the constellation's current capability by looking at the current availability levels of its satellites. Repairs are considered at the satellite functional level, where only one of the satellite's multiple functions may fail or be repaired at any instant in time. In order to incorporate the idea of spare satellites into the methodology, the model would have to be changed to allow simultaneous failures and repairs to occur.

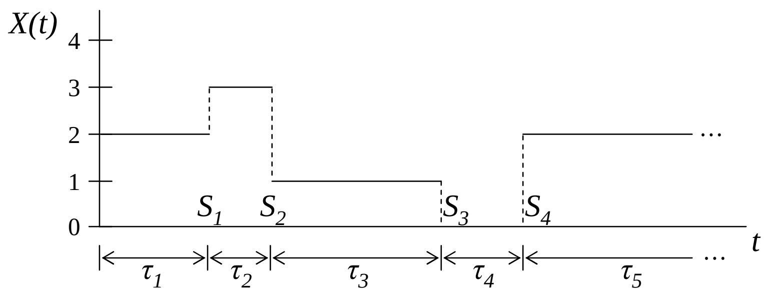

Figure 3.1 Sample path of a discrete state space CTMC.

In order to obtain an instantaneous availability measure for a satellite constellation, we must describe the constellation in terms of a stochastic model. A stochastic process is defined by Kulkarni as "a probabilistic model of a system that evolves randomly" [30:2]. A stochastic process may be described as a sequence of random variables $\{X(t) : t \geq 0\}$ taking on values from a sample space S with $X(t) \in S$ for each $t \geq 0$. The number of elements in set S is referred to as the cardinality of S and is denoted by $|S|$. The cardinality of S may be finite or infinite. Stochastic processes may assume discrete or continuous values over discrete or continuous time intervals. For certain types of stochastic processes, information such as the probability the system will be in a particular state in S at a specific time t, or the long term proportion of time it spends in a given state, can be obtained from the system.

The probability distributions of the duration of time the process remains in its present state, referred to as the *sojourn time*, and the dynamics of state transitions, distinguish certain types of stochastic processes from others. A continuous-time Markov chain (CTMC), denoted $\{X(t) : t \geq 0\}$, is a specific stochastic process which describes the random evolution of a system over continuous time having a countable state space.

Figure 3.1 shows a graphical representation of a sample path of a CTMC, where τ_i is the sojourn time in state i. In the figure, the system starts out in state 2 ($X(0) = 2$) and the process remains in that state for a duration of τ_1 time units until the process makes an instantaneous transition to state 3 at time S_1 ($X(S_1^+) = 3$).

The CTMC process possesses the Markov property at each point of transition, where

$$P\{X(S_{n+1}) = j | X(S_n) = i_n, X(S_{n-1}) = i_{n-1}, \ldots, X(0) = i\}$$
$$= P\{X(S_{n+1}) = j | X(S_n) = i_n\}.$$

A stochastic process is Markovian if the probability of transitioning to a future state only depends on information from the current state. The process remains in a given state for a random amount of time, which is distributed exponentially. The CTMC is time-homogeneous if

$$P\{X(S_{n+1}) = j | X(S_n) = i_n, X(S_{n-1}) = i_{n-1}, \ldots, X(0) = i\}$$
$$= P\{X(S_{n+1}) = j | X(0) = i\}.$$

This implies that all that is needed to find the probability the process transitions to state j is to know the initial state i the process started in. Further information on CTMCs is given in Kulkarni [30].

In this work, we are modelling the stochastic degradation of a satellite of a constellation, which randomly evolves based on the availability of its $M^{(k)}$ functions. Define $X_m^{(k)}(t)$ as a random variable which describes the state of function m for satellite k at time t, $m = 1, 2, \ldots, M^{(k)}$, $k = 1, 2, \ldots, K$, where t is the time of inspection. For the development of the constellation's instantaneous availability, the superscript index will indicate which satellite in the constellation we are referring to and the subscript index will indicate which function of the satellite we are referring to. The random variable $X_m^{(k)}(t)$ is defined as

$$X_m^{(k)}(t) = \begin{cases} 1, & \text{if function } m \text{ is available on satellite } k \text{ at time } t \\ 0, & \text{if function } m \text{ is not available on satellite } k \text{ at time } t \end{cases}. \tag{3.2}$$

In the real world, an unavailable function is not necessarily a failed function but for this methodology, we will assume that a function that is in a failed state is synonymous with a function that is in an unavailable state.

For modeling the state of the satellite, the states of each function of the satellite can be grouped into a row vector $\mathbf{X}^{(k)}(t)$ which represents the state of satellite k at time t:

$$\mathbf{X}^{(k)}(t) = \left[\begin{array}{cccc} X_1^{(k)}(t) & X_2^{(k)}(t) & \ldots & X_M^{(k)}(t) \end{array} \right].$$

Let S be defined as the state space of a satellite, a finite set containing the possible states the satellite may assume at any point in time. The random vector $\mathbf{X}^{(k)}(t)$ will take on a value from set S for each $t \geq 0$. The cardinality of S is $|S| = 2^{M^{(k)}}$, since this represents a Bernoulli trial for each of the $M^{(k)}$ distinct satellite functions.

Each satellite is assumed to be observable over a continuous interval of time, where it may only transition between two states in S in any instant of time. With this model, multiple functions may not fail at the same time; for repairs, multiple functions cannot transition from a failed status to an available status at the same time. This model does not imply that repairs cannot be made concurrently, only that repairs cannot be completed simultaneously. These model characteristics allow the satellite to be modeled as a CTMC process, denoted $\{\mathbf{X}^{(k)}(t), t \geq 0\}$.

Transitions from one state in S to another are based on the discrete event of a satellite either having an available function fail or having an unavailable function repaired. For example, a satellite with $M = 3$ functions will have a state space with $2^3 = 8$ elements given by

$$S = \left\{ \begin{array}{c} (1,1,1),(1,0,1),(0,0,1),(1,0,0), \\ (0,1,1),(1,1,0),(0,1,0),(0,0,0) \end{array} \right\}.$$

Table 3.1 shows an assignment of an index to each element in this state space.

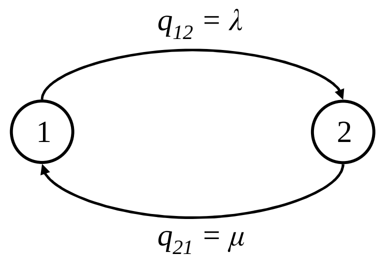

Figure 3.2 Transition rate diagram for a two-state CTMC.

Table 3.1 Index assignment for a three function satellite.

State	Combination	Status
1	(1,1,1)	All functions available
2	(0,1,1)	Function 1 unavailable
3	(1,0,1)	Function 2 unavailable
4	(1,1,0)	Function 3 unavailable
5	(0,0,1)	Functions 1 and 2 unavailable
6	(0,1,0)	Functions 1 and 3 unavailable
7	(1,0,0)	Functions 2 and 3 unavailable
8	(0,0,0)	All functions unavailable

A CTMC can also be represented graphically with a transition rate diagram, a directed graph where each node of the graph represents an element of S and each directed arc shows a possible transition that may occur. Each directed arc has a rate q_{ij} corresponding to the rate at which the process transitions from state i to state j. Figure 3.2 shows an example of a transition rate diagram for a two-state CTMC. This two-state CTMC is often referred to as an up-down machine. Figure 3.3 shows the transition rate diagram corresponding to the index assignment from Table 3.1 along with all of the possible transitions that could occur between the satellite states.

The transition probability matrix, denoted $\mathbf{P}(t)$, is a matrix of probabilities associated with the CTMC where $p_{ij}(t)$ is the (i, j) entry of matrix $\mathbf{P}(t)$. The element $p_{ij}(t)$ indicates the probability of the process transitioning from state i to state j in

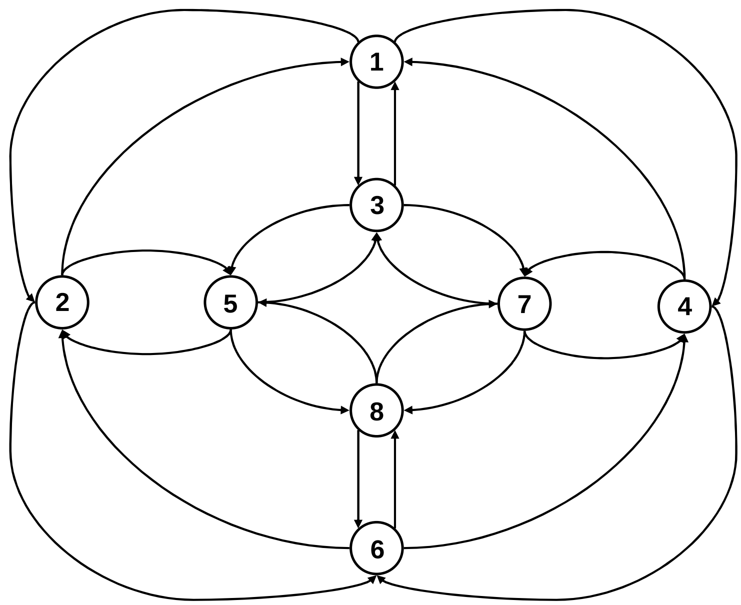

Figure 3.3 Transition rate diagram for a three-function satellite.

t time units, and is defined as

$$p_{ij}(t) = P\{X(t) = j | X(0) = i\}, \quad i, j \in S$$

for a time-homogenous CTMC. We seek the probability distribution associated with the stochastic process $\{X(t), t \geq 0\}$ being in each of the states of the sample space. Let $\alpha_j(t)$ denote the probability that the process is in state j at time t, defined as

$$\alpha_j(t) = P\{X(t) = j\}, \quad j \in S.$$

The probability distribution for the CTMC , denoted $\boldsymbol{\alpha}(t)$, is a row vector of dimension $|S|$, where

$$\boldsymbol{\alpha}(t) = \left[\begin{array}{cccc} \alpha_1(t) & \alpha_2(t) & \ldots & \alpha_{|S|}(t) \end{array} \right].$$

The row vector $\boldsymbol{\alpha}(0)$ is the initial probability distribution which corresponds to the probabilities associated with the process being in each of the possible states in S at

time $t = 0$, where

$$\boldsymbol{\alpha}(0) = \left[\begin{array}{cccc} \alpha_1(0) & \alpha_2(0) & \ldots & \alpha_{|S|}(0) \end{array} \right].$$

An important theorem in [30] and [23] states that a time-homogeneous CTMC is completely described by its initial probability distribution and its transition probability matrix. The probability mass function $\boldsymbol{\alpha}(t)$ for a time-homogeneous CTMC is obtained by conditioning on the probability that the process is in state i at time $t = 0$, where the elements are computed from

$$\alpha_j(t) = \sum_{i \in S} P\{X(t) = j | X(0) = i\} \cdot P\{X(0) = i\},$$

resulting in the probability distribution

$$\boldsymbol{\alpha}(t) = \boldsymbol{\alpha}(0) \cdot \mathbf{P}(t). \tag{3.3}$$

Let $\mathbf{P}^{(k)}(t)$ denote the transition probability matrix for satellite k where $p_{ij}^{(k)}(t)$ is the (i, j) entry of matrix $\mathbf{P}^{(k)}(t)$ and indicates the probability of satellite k transitioning from state i to state j in t time units. The elements of $\mathbf{P}^{(k)}(t)$ are defined as

$$p_{ij}^{(k)}(t) = P\{\mathbf{X}^{(k)}(t) = j | \mathbf{X}^{(k)}(0) = i\} \;\; i, j \in S,$$

since the satellite is modeled as a time-homogenous CTMC. Let $\boldsymbol{\alpha}^{(k)}(t)$ denote the probability distribution of the status of satellite k at time t and let $\boldsymbol{\alpha}^{(k)}(t_0^{(k)})$ be the initial probability distribution which contains the probabilities associated with satellite k being in each of the possible states in S at time $t = t_0^{(k)}$.

$$\boldsymbol{\alpha}^{(k)}(t_0^{(k)}) = \left[\begin{array}{cccc} \alpha_1^{(k)} & \alpha_2^{(k)} & \ldots & \alpha_{|S|}^{(k)} \end{array} \right],$$

where

$$\alpha_i^{(k)} = P\{\mathbf{X}^{(k)}(t_0^{(k)}) = i\} \;\; i \in S.$$

For example, for satellite $k = 4$ with $M^{(4)} = 3$ functions,

$$\boldsymbol{\alpha}^{(4)}(t_0^{(4)}) = \left[\begin{array}{cccccccc} 0.5 & 0.5 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right]$$

describes satellite 4 starting at time $t_0^{(4)}$ having an equal probability of being in states 1 and 2 and having no probability of being in states 3 through 8. If a checkout of satellite k at time $t_0^{(k)}$ reveals that all of its functions are available, then $\boldsymbol{\alpha}^{(k)}(t_0^{(k)})$ will result in a row vector with the first element being equal to one and the remaining elements being equal to zero.

The probability mass function $\boldsymbol{\alpha}^{(k)}(t)$ of $\{\mathbf{X}^{(k)}(t), t \geq 0\}$ is then given by Eqn. (3.3) by conditioning on the probability that satellite k is in state i at time $t = t_0^{(k)}$, given as

$$\boldsymbol{\alpha}^{(k)}(t) = \boldsymbol{\alpha}^{(k)}(t_0^{(k)}) \cdot \mathbf{P}^{(k)}(t). \tag{3.4}$$

Let $\mathbf{Q}$ denote a square matrix of dimension $|S|$ which contains the transition rates from one state to another within set S; q_{ij} is the (i, j) entry of matrix $\mathbf{Q}$ and represents the rate of transition from state i to state j within the $|S|$ states $(i \neq j)$. The matrix $\mathbf{Q}$ is called the infinitesimal generator matrix. The elements of $\mathbf{Q}$ are defined by

$$q_{ij} = \begin{cases} -\sum_{j \neq i} q_{ij}, & i = j \\ q_{ij}, & i \neq j \end{cases} \tag{3.5}$$

where the ith diagonal entry is denoted q_i. The negative of the diagonal elements of $\mathbf{Q}$ are the rate parameters for the sojourn time distributions. Since CTMCs in this work are assumed to be time-homogenous, the transition rates do not change over time. An important property of the infinitesimal generator matrix cited in [30] is

$$\sum_{j \in S} q_{ij} = 0. \tag{3.6}$$

The transition rates are the same rates listed in the transition rate diagram on each arc. In Figure 3.2, the sojourn time associated with the process remaining in state 1 is exponentially distributed with rate λ. Similarly, the time the process is in state 2 is exponentially distributed with rate μ. The infinitesimal generator matrix corresponding to Figure 3.2 is

$$\mathbf{Q} = \begin{bmatrix} -\lambda & \lambda \\ \mu & -\mu \end{bmatrix}.$$

and the infinitesimal generator matrix corresponding to the transition rate diagram in Figure 3.3 is

$$\mathbf{Q} = \begin{bmatrix} -(\lambda_1+\lambda_2+\lambda_3) & \lambda_1 & \lambda_2 & \lambda_3 & 0 & 0 & 0 & 0 \\ \mu_1 & -(\mu_1+\lambda_2+\lambda_3) & 0 & 0 & \lambda_2 & \lambda_3 & 0 & 0 \\ \mu_2 & 0 & -(\mu_2+\lambda_1+\lambda_3) & 0 & \lambda_1 & 0 & \lambda_3 & 0 \\ \mu_3 & 0 & 0 & -(\mu_3+\lambda_1+\lambda_2) & 0 & \lambda_1 & \lambda_2 & 0 \\ 0 & \mu_2 & \mu_1 & 0 & -(\mu_1+\mu_2+\lambda_3) & 0 & 0 & \lambda_3 \\ 0 & \mu_3 & 0 & \mu_1 & 0 & -(\mu_1+\mu_3+\lambda_2) & 0 & \lambda_2 \\ 0 & 0 & \mu_3 & \mu_2 & 0 & 0 & -(\mu_2+\mu_3+\lambda_1) & \lambda_1 \\ 0 & 0 & 0 & 0 & \mu_3 & \mu_2 & \mu_1 & -(\mu_1+\mu_2+\mu_3) \end{bmatrix},$$

where λ_i is the failure rate of function i, and μ_i is the repair rate of function i. Recall that the negative of the diagonal elements of $\mathbf{Q}$ are the rate parameters for the sojourn time distributions.

The transition rate matrix and the probability transition matrix are related by the following equations, also known as the forward and backward Chapman-Kolmogorov equations

$$\frac{d}{dt}\mathbf{P}(t) = \mathbf{P}(t)\mathbf{Q} = \mathbf{Q}\mathbf{P}(t). \tag{3.7}$$

Eqn. (3.7) is a system of homogeneous, first order linear differential equations with initial condition

$$\mathbf{P}(0) = \mathbf{I}, \tag{3.8}$$

where $\mathbf{I}$ is the identity matrix with dimension $|S|$. By definition of a CTMC, at time $t = 0$ it is a sure event that the system will either be in state 1 or state 2; the

process cannot be in two states at the same time. An exact solution to Eqn. (3.7) can be obtained using methods for solving first order differential equations. Once the transition probability matrix is obtained, we can solve for the probability distribution associated with the process.

Let S' be the subset of S which describes the states of the satellite when it is considered to be available to perform its mission, defined as

$$S' = \left\{\sigma_1, \sigma_2, \ldots, \sigma_{|S'|}\right\} \subset S,$$

where σ_i refers to the ith element of S' according to its index assignment. The instantaneous availability $A^{(k)}\left(t - t_0^{(k)}\right)$ of satellite k initialized at time $t_0^{(k)}$, describes the probability of the satellite being in a state of availability at time t, which is equivalent to the sum of the probabilities of the state of satellite k being equal to the elements in set S';

$$A^{(k)}\left(t - t_0^{(k)}\right) = \sum_{j \in S'} \alpha_j(t - t_0^{(k)}) = \sum_{i=1}^{|S'|} P(\mathbf{X}^{(k)}(t - t_0^{(k)}) = \sigma_i). \tag{3.9}$$

3.2.2 Solution Methods

A review of two methods for solving a system of linear differential equations for the probability transition matrix is given below, one using direct integration, and the other using Laplace transforms. An example of solving a two-state system follows, and [16] provides further explanation.

The first method discusses using direct integration to solve for the probability transition matrix given the rate matrix. We start with the Kolmogorov forward equations in standard form where

$$\frac{d}{dt}\mathbf{P}(t) + -\mathbf{QP}(t) = 0. \tag{3.10}$$

The integrating factor, denoted $\boldsymbol{\rho}(t)$, can be obtained by solving the equation

$$\boldsymbol{\rho}(t) = \exp\left(\int -\mathbf{Q}dt\right),$$

which yields

$$\boldsymbol{\rho}(t) = \exp(-\mathbf{Q}t)$$

since $\mathbf{Q}$ is a constant. Multiplying both sides of Eqn. (3.10) by this integrating factor gives

$$\exp(-\mathbf{Q}t)\left[\frac{d}{dt}\mathbf{P}(t) - \mathbf{Q}\mathbf{P}(t)\right] = 0. \tag{3.11}$$

We can recognize the left side of Eqn. (3.11) as the derivative of a product of the integrating factor $\exp(-\mathbf{Q}t)$ and $\mathbf{P}(t)$. Thus Eqn. (3.11) becomes

$$\frac{d}{dt}\{\exp(-\mathbf{Q}t)\mathbf{P}(t)\} = 0.$$

When we integrate both sides with respect to t,

$$\exp(-\mathbf{Q}t)\mathbf{P}(t) = \mathbf{C}, \tag{3.12}$$

where $\mathbf{C}$ is a constant, square matrix of dimension $|S|$. From our initial condition, the matrix $\mathbf{P}(0) = \mathbf{I}$. Using this fact, we can solve for $\mathbf{C}$:

$$\mathbf{C} = \exp(-\mathbf{Q}\cdot 0)\mathbf{P}(0) = \mathbf{I}.$$

After substituting $\mathbf{I}$ for $\mathbf{C}$ into Eqn. (3.12) and solving for $\mathbf{P}(t)$, we obtain

$$\mathbf{P}(t) = \exp(\mathbf{Q}t),$$

where $\exp(\mathbf{Q}t)$ is the matrix exponentiation of $\mathbf{Q}t$ and is defined as

$$\exp(\mathbf{Q}t) = \sum_{n=0}^{\infty} \frac{(\mathbf{Q}t)^n}{n!} = \mathbf{I} + \sum_{n=1}^{\infty} \frac{(\mathbf{Q}t)^n}{n!}. \tag{3.13}$$

The use of Laplace transforms is a second method to solve Eqn. (3.7) for $\mathbf{P}(t)$. This involves transforming the system from the time domain to the complex number domain, where the system of ordinary differential equations becomes a system of linear equations that is much easier to solve. After the transform solution is obtained in the transform domain, the system is transformed back to the time domain through an inverse Laplace transform. Let $f(t)$ be an absolutely integrable function on the positive real line and let $f^*(s) = \mathcal{L}\{f(t)\}$ denote the Laplace transform of $f(t)$ given by

$$f^*(s) = \int_0^{\infty} e^{-st} f(t)dt, \quad t \geq 0, \quad s \in \mathbb{C}, \quad \Re(s) > 0, \tag{3.14}$$

where $\mathbb{C}$ denotes the set of complex numbers and $\Re(s)$ denotes the real part of the complex number s. The Laplace transform of the probability transition matrix $\mathbf{P}(t)$ is denoted by $\mathbf{P}^*(s)$, where its (i, j)th element $p^*_{ij}(s)$ is defined as

$$p^*_{ij}(s) = \int_0^{\infty} e^{-st} p_{ij}(t)dt.$$

We will apply the Laplace transform to both sides of Eqn. (3.7). From properties of Laplace transforms [30],

$$\mathcal{L}\left\{\frac{d}{dt}\mathbf{P}(t)\right\} = s\mathbf{P}^*(s) - \mathbf{P}(0).$$

The Laplace transform of a function multiplied by a constant is equal to the constant multiplied by the Laplace transform of the function. In the case of Eqn. (3.7), $\mathbf{Q}$ is a constant:

$$\mathcal{L}\{\mathbf{Q}\mathbf{P}(t)\} = \mathbf{Q}\mathcal{L}\{\mathbf{P}(t)\} = \mathbf{Q}\mathbf{P}^*(s).$$

Applying the Laplace transform to Eqn. (3.7) yields

$$s\mathbf{P}^*(s) - \mathbf{P}(0) = \mathbf{Q}\mathbf{P}^*(s).$$

Using the initial condition $\mathbf{P}(0) = \mathbf{I}$ with factoring yields

$$(s\mathbf{I} - \mathbf{Q})\mathbf{P}^*(s) = \mathbf{I}.$$

After left-multiplying both sides of the equation by $(s\mathbf{I} - \mathbf{Q})^{-1}$, we have

$$\mathbf{P}^*(s) = (s\mathbf{I} - \mathbf{Q})^{-1}. \tag{3.15}$$

Eqn. (3.15) is a matrix of rational functions in s. The inverse Laplace transform can be applied to $\mathbf{P}^*(s)$ to obtain an exact solution via properties of inverse Laplace transforms and partial fraction decomposition [30]. The inverse Laplace transform of $\mathbf{P}^*(s)$, denoted $\mathcal{L}^{-1}\{\mathbf{P}^*(s)\}$, brings the probability transition matrix back into the time domain where

$$\mathbf{P}(t) = \mathcal{L}^{-1}\{\mathbf{P}^*(s)\} = \mathcal{L}^{-1}\{(s\mathbf{I} - \mathbf{Q})^{-1}\}. \tag{3.16}$$

The formal equation for the inverse Laplace transform, also known as the Bromwich or Fourier-Mellin integral, is given by

$$\mathcal{L}^{-1}\{f^*(s)\} = f(t) = \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} e^{st} f^*(s) ds, \quad t \geq 0, \tag{3.17}$$

where $i = \sqrt{-1}$. The exact solution to $\mathbf{P}(t)$ can be obtained by observation, by applying some basic properties of inverse Laplace transforms after partial fraction decomposition has been performed.

The solution obtained using an integrating factor and using the Laplace transform technique yield the same solution. An approximation to $\mathbf{P}(t)$ can be obtained by

applying numerical methods to solve Eqn. (3.13) or Eqn. (3.15). Many matrix-based computer software packages come equipped with matrix exponentiation functions that can be applied to approximate the solution to Eqn. (3.13). For example, MatLab release 13 comes equipped with several variations of a function that performs matrix exponentiation. Numerical algorithms for performing the inverse Laplace transform are numerous and stem from a seminal paper from Dubner and Abate [13].

To calculate the instantaneous availability of a system, a subset must be defined in the system's state space that is equivalent to the system being available. After this subset is defined, transient analysis can be performed to determine the probabilities associated with the system being in those states of availability at time t; the instantaneous availability, $A(t)$, can be determined by summing the probabilities associated with those states. Let S' be the subset of S which describes the states of availability:

$$S' = \left\{\sigma_1, \sigma_2, \ldots, \sigma_{|S'|}\right\} \subset S.$$

The value $A(t)$ is then equal to the sum of the probabilities of the system being in set S' at time t:

$$A(t) = \sum_{j \in S'} \alpha_j(t) = \sum_{i=1}^{|S'|} P(\mathbf{X}(t) = \sigma_i). \tag{3.18}$$

Next, we discuss the CTMC analysis for determining the probability mass function and the instantaneous availability for the two-state CTMC in Figure 3.2, where $S = \{1, 2\}$. For calculating the probability mass function $\boldsymbol{\alpha}(t)$, we need the initial probability distribution vector, the infinitesimal generator matrix, and the desired time t. For calculating the instantaneous availability $A(t)$, we need to know what states in S are indicative of the system being available.

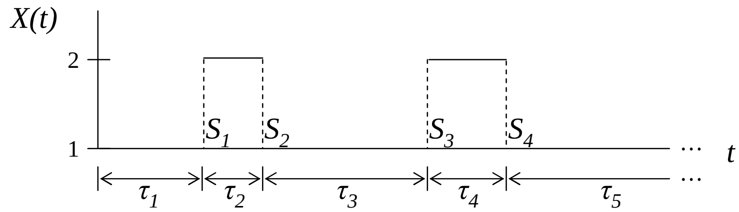

Figure 3.4 Sample path of a two-state CTMC.

Let the system be available when it is in state 1 and let the system start in state 1 at time $t = 0$ by having an initial probability distribution vector

$$\boldsymbol{\alpha}(0) = \begin{bmatrix} 1 & 0 \end{bmatrix}.$$

The system is up when it is in state 1 and down when it is in state 2. Figure 3.4 shows a sample path of an up down machine starting in state 1. Let the system have exponential distributions for its time in each state, where λ and μ refer to the rates of failure and repair, respectively, resulting in an infinitesimal generator matrix

$$\mathbf{Q} = \begin{bmatrix} -\lambda & \lambda \\ \mu & -\mu \end{bmatrix}.$$

We will now solve Eqn. (3.7) for $\mathbf{P}(t)$. The forward Chapman Kolmogorov equation

$$\frac{d}{dt}\mathbf{P}(t) = \mathbf{P}(t)\mathbf{Q}$$

becomes

$$\begin{aligned} \frac{d}{dt}\begin{bmatrix} p_{11}(t) & p_{12}(t) \\ p_{21}(t) & p_{22}(t) \end{bmatrix} &= \begin{bmatrix} p_{11}(t) & p_{12}(t) \\ p_{21}(t) & p_{22}(t) \end{bmatrix}\begin{bmatrix} -\lambda & \lambda \\ \mu & -\mu \end{bmatrix} \\ &= \begin{bmatrix} -\lambda p_{11}(t) + \mu p_{12}(t) & \lambda p_{11}(t) - \mu p_{12}(t) \\ -\lambda p_{21}(t) + \mu p_{22}(t) & \lambda p_{21}(t) - \mu p_{22}(t) \end{bmatrix}, \end{aligned} \tag{3.19}$$

which is equivalent to the linear differential equations

$$\frac{d}{dt}p_{11}(t) = -\lambda p_{11}(t) + \mu p_{12}(t) \tag{3.20}$$
$$\frac{d}{dt}p_{12}(t) = \lambda p_{11}(t) - \mu p_{12}(t) \tag{3.21}$$
$$\frac{d}{dt}p_{21}(t) = -\lambda p_{21}(t) + \mu p_{22}(t) \tag{3.22}$$
$$\frac{d}{dt}p_{22}(t) = \lambda p_{21}(t) - \mu p_{22}(t) \tag{3.23}$$

with initial conditions

$$p_{11}(0) = p_{22}(0) = 1 \text{ and } p_{12}(0) = p_{21}(0) = 0.$$

From an axiom of probability, the rows of $\mathbf{P}(t)$ have to sum to 1:

$$p_{11} + p_{12} = p_{21} + p_{22} = 1. \tag{3.24}$$

Applying this rule of complements allows us to solve for all of the probability values by only solving one probability value from each row of $\mathbf{P}(t)$. We will show the solution for p_{11} and p_{12}.

Using substitution from Eqn. (3.24) and arranging in standard form, Eqn. (3.20) becomes

$$\frac{d}{dt}p_{11}(t) + (\lambda + \mu)p_{11}(t) = \mu. \tag{3.25}$$

Our integrating factor is observed to be

$$\rho(t) = \exp((\lambda + \mu)t),$$

and multiplying both sides of Eqn. (3.25) by this integrating factor yields

$$\exp((\lambda + \mu)t)\left[\frac{d}{dt}p_{11}(t) + (\lambda + \mu)p_{11}(t)\right] = \exp((\lambda + \mu)t)\mu. \tag{3.26}$$

We can recognize the left side of Eqn. (3.26) as the derivative of a product of the integrating factor $\exp((\lambda+\mu)t)$ and $p_{11}(t)$. Thus Eqn. (3.26) becomes

$$\frac{d}{dt}\{\exp((\lambda+\mu)t)p_{11}(t)\} = \exp((\lambda+\mu)t)\mu$$

and integrating both sides with respect to t gives

$$\exp((\lambda+\mu)t)p_{11}(t) = \frac{\mu}{\lambda+\mu}\exp((\lambda+\mu)t) + c. \tag{3.27}$$

From our initial condition, $p_{11}(0) = 1$ can be used to solve Eqn. (3.27) for c resulting in

$$c = \frac{\lambda}{\lambda+\mu}.$$

After substituting the result for c into Eqn. (3.27) and solving for $p_{11}(t)$, we obtain

$$p_{11}(t) = \frac{\mu}{\lambda+\mu} + \frac{\lambda}{\lambda+\mu}\exp(-(\lambda+\mu)t).$$

From Eqn. (3.24),

$$\begin{aligned} p_{12}(t) &= 1 - p_{11}(t) \\ &= 1 - \left[\frac{\mu}{\lambda+\mu} + \frac{\lambda}{\lambda+\mu}\exp(-(\lambda+\mu)t)\right] \\ &= \frac{\lambda}{\lambda+\mu} - \frac{\lambda}{\lambda+\mu}\exp(-(\lambda+\mu)t). \end{aligned}$$

In a similar manner, solutions for equations (3.22) and (3.23) can be obtained resulting in the matrix

$$\mathbf{P}(t) = \begin{bmatrix} \frac{\mu}{\lambda+\mu} + \frac{\lambda}{\lambda+\mu}\exp(-(\lambda+\mu)t) & \frac{\lambda}{\lambda+\mu} - \frac{\lambda}{\lambda+\mu}\exp(-(\lambda+\mu)t) \\ \frac{\mu}{\lambda+\mu} - \frac{\mu}{\lambda+\mu}\exp(-(\lambda+\mu)t) & \frac{\lambda}{\lambda+\mu} + \frac{\mu}{\lambda+\mu}\exp(-(\lambda+\mu)t) \end{bmatrix}.$$

The probability mass function can be obtained by multiplying the initial probability distribution to $\mathbf{P}(t)$, resulting in

$$\begin{aligned}
\boldsymbol{\alpha}(t) &= \boldsymbol{\alpha}(0)\mathbf{P}(t) \\
&= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \frac{\mu}{\lambda+\mu} + \frac{\lambda}{\lambda+\mu}\exp\left(-(\lambda+\mu)t\right) & \frac{\lambda}{\lambda+\mu} - \frac{\lambda}{\lambda+\mu}\exp\left(-(\lambda+\mu)t\right) \\ \frac{\mu}{\lambda+\mu} - \frac{\mu}{\lambda+\mu}\exp\left(-(\lambda+\mu)t\right) & \frac{\lambda}{\lambda+\mu} + \frac{\mu}{\lambda+\mu}\exp\left(-(\lambda+\mu)t\right) \end{bmatrix} \\
&= \begin{bmatrix} \frac{\mu}{\lambda+\mu} + \frac{\lambda}{\lambda+\mu}\exp\left(-(\lambda+\mu)t\right) & \frac{\lambda}{\lambda+\mu} - \frac{\lambda}{\lambda+\mu}\exp\left(-(\lambda+\mu)t\right) \end{bmatrix}.
\end{aligned}$$

The availability of the system is calculated by summing all of the probabilities of the states which indicate the system is available. In this case, the system is available when it is in state 1. Therefore,

$$A(t) = \frac{\mu}{\lambda+\mu} + \frac{\lambda}{\lambda+\mu}\exp\left(-(\lambda+\mu)t\right)$$

is the probability the system will be available at time t with failure and repair rates λ and μ, respectively.

3.3 Limiting Availability

Methods for obtaining the limiting availability for a system modeled as a CTMC are given below for comparison purposes. Chapter 5 will present numeric examples with calculations of both the limiting availability and the instantaneous availability to show that these methods produce significantly different results when assessing a constellation's functional capability.

As was previously stated, limiting availability is a long-run average of the proportion of time a system is available over its entire life span. Given the instantaneous availability $A(t)$, the limiting availability A is defined as

$$A = \lim_{t\to\infty} A(t).$$

In terms of satellite availability, let $A^{(k)}$ and $A^{(k)}(t)$ denote the limiting availability and the instantaneous availability for satellite k, respectively. Then

$$A^{(k)} = \lim_{t\to\infty} A^{(k)}(t) = \lim_{t\to\infty} \sum_{i=1}^{|S'|} P(\mathbf{X}^{(k)}(t) = \sigma_i).$$

The methods for obtaining the limiting availability for a system modeled as a CTMC consist of obtaining the probability distribution for the satellite when it has reached a limiting state of equilibrium, then summing up the long-term probabilities associated with states of availability. Let $\mathbf{p}^{(k)}$ denote a row vector consisting of probabilities $p_i^{(k)}$ defined as

$$p_i^{(k)} = \lim_{t\to\infty} P(\mathbf{X}^{(k)}(t) = \sigma_i), \quad \sigma_i \in S, \quad i = 1, 2, \ldots, |S|.$$

The probability $p_i^{(k)}$ represents the long term probability the satellite is found in state i and can also be view as the proportion of time the system spends in state i over its entire life. The system of equations

$$\begin{aligned} \mathbf{p}^{(k)}\mathbf{Q}^{(k)} &= \mathbf{0} \\ \mathbf{p}^{(k)}\mathbf{e} &= 1 \end{aligned}$$

is solved for $\mathbf{p}^{(k)}$, where $\mathbf{Q}^{(k)}$ is the rate matrix associated with satellite k, $\mathbf{0}$ denotes a row vector of zeros and $\mathbf{e}$ denotes a row vector of ones. The first equation implies the condition of the satellite being in a state of equilibrium where the rate of probability flow from one state to another is equal to zero. The second equation is the normalizing condition, corresponding to the axioms of probability. The limiting availability can then be obtained by summing the probabilities associated with states of availability where

$$A^{(k)} = \sum_{i=1}^{|S'|} p_i^{(k)}.$$

3.4 Satellite Value Score

This section gives an overview of one possible methodology for creating the satellite value score $V^{(k)}(t)$, the value of satellite k to support the overall constellation mission at time t. This measure is included to account for scenarios in which a satellite may be functioning at its specified design but may not be contributing to the current mission of the satellite constellation. This situation may occur due to problems associated with the acquisition of the satellite, such as a lengthy acquisition schedule resulting in the fielding of a satellite with obsolete technology or a satellite that no longer meets mission requirements due to a dynamic threat environment.

For this particular method, the satellite value score is obtained through the use of multi-attribute value theory, which describes the methodology of assigning numerical value to a subject through the direct evaluation of quantitative attribute scores or through the scoring of qualitative attributes by a subject matter expert (SME). In this application, the satellite value score is a normalized quantity that represents the opinion of a SME on the operational value of a satellite, where the attributes reflect a satellite's contribution to the overall constellation mission.

The value score can take on many different forms. An additive form for this value is given in [7] and [29] and is computed for each satellite as

$$V^{(k)}(t) = \sum_{i=1}^{N} w_i^{(k)}(t) v_i^{(k)}(y_i), \quad \sum_{i=1}^{N} w_i^{(k)}(t) = 1, \tag{3.28}$$

where N denotes the number of attributes used to score the satellite, $w_i^{(k)}(t)$ is the normalized weighting factor for attribute i of satellite k at time t, and $v_i^{(k)}(y_i)$ is the value function of attribute i for satellite k evaluated with score y_i. In order to use Eqn. (3.28) to obtain a value score, the attributes are assumed to meet the condition of additive independence given in [7]. Obtaining the value score requires first determining which attributes to include, creating scales for qualitative attributes,

determining an appropriate value function for each attribute which maps the attribute scale to a normalized attribute value, then rank-ordering and assigning a weight of importance to each attribute. The value score will be obtained by summing the product of the attribute scores and the attribute weights. The input of a SME familiar with both the operation of the satellites and the mission that depends on the constellation will be critical to establishing meaningful satellite value scores.

3.4.1 Attribute Value Functions

The first step of this methodology involves determining a set of applicable attributes that reflects the satellite's ability to fulfill current mission requirements. An attribute can be quantitative or qualitative. For attributes of a quantitative nature such as satellite performance metrics, an appropriate range must be determined to create the attribute scale. For attributes of a qualitative nature, a categorical scale should be created to differentiate between different levels of the attribute, where a numeric scale can be created from the worst and best subjective categories. The attribute scales are generated to create a domain for the attribute value functions.

For example, in [35], an attribute describing a satellite's contribution to the parent constellation mission is assigned a scale from 0 to 1, where the satellite is given an attribute score of 0.0 if it is a spare or does not offer any support to the constellation mission, 0.25 if it only has secondary impact on the constellation mission, 0.5 if it generally impacts the constellation mission, 0.75 if it strongly impacts the constellation mission, and 1.0 if it is critical to the constellation mission. In this particular case, the attribute scale is normalized but this is not a necessary requirement.

Next, the value function, denoted $v_i^{(k)}(y_i)$, is obtained for attribute i of satellite k. The value function is created with the help of a SME and yields a normalized value representing the satellite's value relative to the attribute. After value functions

are created for each attribute, the satellite attributes can be scored by a SME using interview questions, where y_i denotes the score for attribute i.

For example, given attributes with normalized scales, a value function of an exponential form can be created by the SME determining where on the attribute scale a value score of 0.5 would correspond to. Based on this measure, denoted here as y_i', $v_i^{(k)}(y_i') = 0.5$. The exponential value function would then be given as

$$v_i^{(k)}(y_i) = \frac{1 - \exp(c_i^{(k)} y_i)}{1 - \exp(c_i^{(k)})}, \tag{3.29}$$

where $c_i^{(k)}$ is a constant that can be computed by applying numerical techniques such as the bisection method or Newton's method to solve the equation

$$0.5 = \frac{1 - \exp(c_i^{(k)} y_i')}{1 - \exp(c_i^{(k)})}. \tag{3.30}$$

The constant $c_i^{(k)}$ is referred to in [35] as the risk attitude constant (RAC). Regarding the SME's chosen value for y_i', the value function is linear if $y_i' = 0.5$, concave if $y_i' < 0.5$, and convex if $y_i' > 0.5$.

For attribute scales which may not be normalized, a value function of the form

$$v_i^{(k)}(y_i) = \frac{1}{1 + a_i^{-b_i y_i}} \tag{3.31}$$

may be used, where a_i and b_i are constants associated with attribute i. Eqn. (3.31) is referred by [11] as an S-shaped curve. In order to solve for these constants, more than one score on the attribute scale representing value must be solicited from an SME (i.e., multiple scores of y_i'). The constants a_i and b_i can then be solved for using numerical methods.

3.4.2 Attribute Weights

Once every attribute has a corresponding value function, attribute weights can be assigned, allowing preference between attributes to be accounted for in the satellite value score. As stated earlier, $w_i^{(k)}(t)$ refers to the normalized weighting factor of attribute i for satellite k at the time of evaluation of the constellation. This measure is time-dependent to account for changes in the order of preference of attributes for constellation missions of a dynamic nature.

Normalized attribute weights are necessary components of the satellite value score and methods for determining these weights have been studied extensively in the literature. For completeness, we point out three methods described in [7] for assessing attribute weights: the pricing-out method, the swing-weight approach, and the lottery-comparison technique. A more mathematical explanation of the swing-weight approach and the pricing-out method is given in [29], where attribute weights are referred to as scaling constants.

3.5 Model Summary

This chapter showed the formulation of the constellation functional capability measure by first introducing the equation for the measure, giving a review of CTMC analysis, then explaining the two components of the capability measure in terms of an individual satellite's instantaneous availability and its functional value score. A model was given for satellite instantaneous availability where the satellites are modeled as time-homogeneous CTMCs corresponding to exponential failure and repair distributions. Next, a method for computing the limiting availability was given for comparison purposes. Finally, the satellite value score was derived, giving an example of one methodology that may be used to assign a value to a satellite, based on criteria determined by a subject matter expert, which reflects the capability of a satellite fulfilling a specific mission requirement. The next chapter will discuss

an extension to the methodology presented in this chapter for computing the instantaneous capability of a satellite constellation given its function lifetimes are not exponentially distributed.

4. Semi-Markov Degradation Process

This chapter describes an alternate (and more general) method for computing the instantaneous capability of a constellation containing satellites with generally distributed failure and repair times. The methodology is similar to that of Chapter 3, however, the satellite's instantaneous availability is acquired by modeling its degradation status as a time-homogenous semi-Markov process (SMP) where the failure and repair time distributions are not restricted to exponential distributions. The satellite value scores are computed in the same manner as described in the previous chapter.

4.1 Semi-Markov Process (SMP) Model

Recall that $\mathbf{X}^{(k)}(t)$ is the random vector describing the state of satellite k at time t, where each element in the vector indicates the state of availability of the specific functions of the satellite. State space S contains all of the combinations of states satellite k may assume. The stochastic process $\{\mathbf{X}^{(k)}(t) : t \geq 0\}$ evolves randomly assuming values in state space S.

In order to obtain the constellation's functional capability measure, each satellite's instantaneous availability must be obtained. The instantaneous availability measure for a satellite constellation that has non-exponential failure and repair distributions must be described in terms of a semi-Markov process (SMP). Modeling the satellite as a SMP requires knowledge of the distributions of the state holding times of the satellite and the transition probabilities associated with each of the states in S. The following assumptions apply to the SMP model of satellite degradation:

1. The constellation contains a finite number of independent satellites, each with a finite set of functions.

2. The transition rates between states of availability are assumed to be estimable from observed data.

3. The probability distributions of sojourn durations in individual states are known or may be estimated (parametrically).

4. Multiple function failures or repairs cannot occur in any instant of time.

5. The initial state of the satellite constellation is assumed to be known; the states associated with the different combinations of states the satellite may be in are assumed to be definable and the probabilities associated with the satellite being in these states when the satellite first came on line are assumed to be known.

6. The availability status of the satellite can be determined based on the availability of specific functions.

A short review of SMPs is given before the SMP degradation model is presented. A description of renewal processes including the more specific Markov renewal processes is first given, as SMPs are defined by an embedded Markov renewal process. Further information on SMPs is given in Kulkarni [30].

A renewal process is a counting process which counts the number of events from a renewal sequence. Let S_n be the time of the nth event and let $N(t)$ be the number of events up to time t. If the times between events are independent and identically distributed random variables, then $\{S_n : n \geq 0\}$ is a renewal sequence and $\{N(t) : t \geq 0\}$ is a renewal process [30]. Similarly, a Markov renewal process counts the number of events from a Markov renewal sequence. A Markov renewal sequence is a sequence of bivariate random variables $\{(Y_n, S_n) : n \geq 0\}$ where Y_n indicates the observation of the process at the nth transition epoch S_n. Let $\{X(t) : t \geq 0\}$ denote a continuous-time stochastic process where $X(t) \in S$, and let X_n denote the state of the process after the nth transition epoch S_n, where $X_n = X(S_n^+)$. From [30], the

sequence $\{(X_n, S_n) : n \geq 0\}$ is a Markov renewal sequence if

$$
\begin{aligned}
&P\{X_{n+1} = j, S_{n+1} - S_n \leq t | X_n = i, X_{n-1}, \ldots, X_0, S_{n-1}, S_{n-2}, \ldots, S_0\} \\
&\quad = P\{X_{n+1} = j, S_{n+1} - S_n \leq t | X_n = i\} \text{ (Markov Property).} \qquad (4.1)
\end{aligned}
$$

The Markov renewal sequence is *time-homogeneous* if

$$
P\{X_{n+1} = j, S_{n+1} - S_n \leq t | X_n = i\} = P\{X_1 = j, S_1 \leq t | X_0 = i\}. \qquad (4.2)
$$

The Markov property implies that the probability that the process transitions from state i to state j at any transition epoch can be determined by knowing the <u>current</u> state, regardless of the history, along with the sojourn time distribution in state i. Time homogeneity implies that these transition probabilities and associated sojourn time distributions do not change with time.

A continuous-time stochastic process $\{X(t) : t \geq 0\}$ is a SMP if it has an embedded Markov renewal sequence $\{(X_n, S_n) : n \geq 0\}$ where X_n is defined to be equal to $X(S_n^+)$, the state after the nth transition epoch S_n. An SMP is piecewise constant and right-continuous with left-hand limits everywhere. Figure 4.1 shows a sample path of a SMP, along with its embedded Markov renewal sequence. The process starts at time epoch S_0 in state Y_0, where $S_0 = 0$, and is observed again at time epoch S_1 to be in state Y_1. The process continues to evolve and is observed at discrete time epochs.

4.2 Instantaneous Availability

In order for the instantaneous availability to be calculated for a satellite degradation process modeled as a SMP, the sojourn time distributions for each state of the sample space S must be known as well as the state transition probabilities.

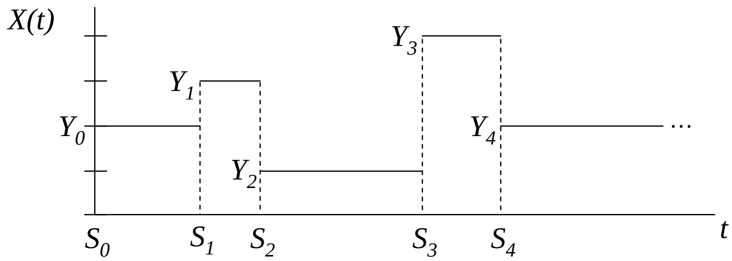

Figure 4.1 Sample path of a SMP.

For this chapter, the probability mass function $\boldsymbol{\alpha}(t)$ is defined as it was for Chapter 3. Recall that $\alpha_j(t)$ denotes the probability that the process is in state j at time t, defined as

$$\alpha_j(t) = P\{X(t) = j\}, \quad j \in S,$$

and the probability mass function $\boldsymbol{\alpha}(t)$ is a row vector of dimension $|S|$, where

$$\boldsymbol{\alpha}(t) = \left[\begin{array}{cccc} \alpha_1(t) & \alpha_2(t) & \ldots & \alpha_{|S|}(t) \end{array} \right].$$

The row vector $\boldsymbol{\alpha}(0)$ is the initial probability distribution which corresponds to the probabilities associated with the process being in each of the possible states in S at time $t = 0$, where

$$\boldsymbol{\alpha}(0) = \left[\begin{array}{cccc} \alpha_1(0) & \alpha_2(0) & \ldots & \alpha_{|S|}(0) \end{array} \right].$$

The matrix $\mathbf{G}(t)$, consisting of elements

$$G_{ij}(t) = P\{X_1 = j, S_1 \leq t | X_0 = i\} \tag{4.3}$$

defined by Eqn. (4.2), is referred to as the semi-Markov kernel. A SMP is completely described by its initial probability distribution $\boldsymbol{\alpha}(0)$ and its semi-Markov kernel $\mathbf{G}(t)$, [30].

The semi-Markov kernel is directly related to the sojourn time distributions of the SMP. Let T_{ij} denote the random amount of time the process stays in state i

given that it next transitions to state j. The random variable T_{ij} has a c.d.f. $H_{ij}(t)$ where

$$H_{ij}(t) = P\{S_1 \le t | X_0 = i, X_1 = j\}.$$

Let T_i denote the random (unconditional) amount of time the process is in state i. We seek the c.d.f $H_i(t)$ for the random variable T_i. By conditioning on the possible states to which the process may transition, not including state i,

$$\begin{aligned} H_i(t) &= \sum_{j \in S} H_{ij}(t) P\{X_1 = j\} \\ &= \sum_{j \in S} P\{S_1 \le t | X_0 = i, X_1 = j\} P\{X_1 = j\} \\ &= \sum_{j \in S} P\{X_1 = j, S_1 \le t | X_0 = i\}. \end{aligned}$$

Therefore,

$$H_i(t) = \sum_{j \in S} G_{ij}(t), \tag{4.4}$$

which indicates that the sojourn time distributions are composed of the row sums of the semi-Markov kernel.

The semi-Markov kernel can be obtained using the conditional state sojourn time distributions and the transition probabilities as

$$\begin{aligned} G_{ij}(t) &= P\{X_1 = j, S_1 \le t | X_0 = i\} \\ &= P\{X_1 = j | X_0 = i\} P\{S_1 \le t | X_0 = i, X_1 = j\} \\ &= p_{ij} H_{ij}(t). \end{aligned} \tag{4.5}$$

Solving for the semi-Markov kernel matrix in this manner requires knowledge of the c.d.f. associated with the sojourn time in state i given that the process will transition to state j. This also requires the transition probability associated with the process transitioning from state i to state j.

The data required to apply this methodology to a satellite's degradation process is difficult to obtain in practice and as such, we make the assumption that the sojourn time of a given state is independent of the subsequent state so that

$$H_i(t) = H_{ij}(t), \quad \text{for all } j \neq i.$$

With this assumption, we may write

$$G_{ij}(t) = p_{ij} H_i(t). \tag{4.6}$$

However, as more historical data is collected regarding the distributions, the model may be updated to reflect the sojourn time distributions for specific state transitions. One possible way to incorporate new distribution information is as follows. Let $|S|-1$ denote the total number of states to which the process can transition, not including the current state. In general, the state sojourn time distribution is given as

$$\begin{aligned} H_i(t) &\equiv P\{T_i \leq t\} \\ &= 1 - P\{T_i > t\} \\ &= 1 - P\{\min(T_{i,1}, T_{i,2}, \ldots, T_{i,|S|-1}) > t\} \\ &= 1 - P\{T_{i,1} > t, T_{i,2} > t, \ldots, T_{i,|S|-1} > t\}. \end{aligned} \tag{4.7}$$

Since the conditional sojourn time distributions are independent,

$$H_i(t) = 1 - \prod_{j \neq i} \overline{H}_{ij}(t), \tag{4.8}$$

where $\overline{H}_{ij}(t)$ is the complement of $H_{ij}(t)$.

In the case of the CTMC model, all of the sojourn time distributions are exponential; $H_i(t)$ is simply the distribution of the minimum of exponentially distributed random variables which is exponential with rate parameter equal to the sum of the

rates associated with the random variables $T_{i,1}, T_{i,2}, \cdots, T_{i,|S|-1}$. The rates for the state holding time distributions for a CTMC correspond to the diagonal elements of the rate matrix $\mathbf{Q}$, where the negative of the ith diagonal element corresponds to the rate of the ith state holding time distribution. The resulting semi-Markov kernel matrix has elements

$$G_{ij}(t) = \begin{cases} \frac{q_{ij}}{q_i}(1 - e^{-q_i t}), & i \neq j \\ 0, & i = j \end{cases}. \tag{4.9}$$

In the case of the SMP model, the state holding times may not be distributed exponentially; therefore the transition probabilities p_{ij} may not be known. One way to approximate the semi-Markov kernel is to approximate the transition probabilities by assuming that a statistical sample of the transition rates can estimate the transition probability as if the distributions of the state holding times are exponential. With estimates for the transition rates, we can approximate the semi-Markov kernel, solve a system of equations for $\mathbf{P}(t)$, and find the instantaneous availability by using Eqn. (3.18).

Let r_{ij} denote a statistical estimator for the rate of transition from state i to state j given by

$$r_{ij} = \frac{N_{ij}(T)}{T}, \tag{4.10}$$

where T is the observation period and $N_{ij}(T)$ is the random number of transitions from state i to state j in time period T. Note that the observed number of transitions and the period of observation need to be sufficiently large to provide a reasonable statistical estimator for the true rate of transition [39]. Also, we may take as the approximate rate of transition out of state i

$$r_i = -r_{ii} = \sum_{j \neq i} r_{ij}. \tag{4.11}$$

From Eqn. (4.10) and Eqn. (4.11), we can construct a matrix $\mathbf{R}$ which represents an estimator for the rate matrix $\mathbf{Q}$ where

$$r_{ij} \approx q_{ij} \text{ and } r_i \approx q_i.$$

The semi-Markov kernel can now be approximated by applying the above statistics to Eqn. (4.6) where

$$\widehat{G}_{ij}(t) = \begin{cases} \frac{r_{ij}}{r_i} H_i(t), & i \neq j \\ 0, & i = j. \end{cases} \tag{4.12}$$

Let the matrix $\mathbf{D}(t)$ be a diagonal matrix with elements

$$D_{ij}(t) = \begin{cases} \overline{H}_i(t), & i = j \\ 0, & i \neq j \end{cases}. \tag{4.13}$$

The following equation characterizes the relationship between $\mathbf{P}(t)$, $\mathbf{G}(t)$, and $\mathbf{D}(t)$ [30]:

$$\mathbf{P}(t) = \mathbf{D}(t) + \mathbf{G} * \mathbf{P}(t), \tag{4.14}$$

where

$$\mathbf{G} * \mathbf{P}(t) = \int_0^t \mathbf{G}(t-u) d\mathbf{P}(u) \tag{4.15}$$

is the convolution of $\mathbf{G}(t)$ with $\mathbf{P}(t)$.

Let $f(t)$ be a function on the positive real line and let $\widetilde{f}(s)$ denote the Laplace-Stieltjes transform (LST) of $f(t)$ where s is a complex variable. The Laplace-Stieltjes transform is defined as

$$\widetilde{f}(s) = \int_0^\infty e^{-st} df(t), \quad t \geq 0, \quad s \in \mathbb{C}, \quad \Re(s) > 0. \tag{4.16}$$

The Laplace-Stieltjes transform of matrix $\mathbf{P}(t)$ will be denoted $\widetilde{\mathbf{P}}(s)$ where its (i, j)th element is $\widetilde{p}_{ij}(s)$.

An important theorem relative to Laplace-Stieltjes transforms and the convolution of two functions states that given two functions $F(t)$ and $G(t)$,

$$LST\{F * G(t)\} = \widetilde{F}(s)\widetilde{G}(s).$$

By taking the LST of both sides of Eqn. (4.14),

$$\widetilde{\mathbf{P}}(s) = \widetilde{\mathbf{D}}(s) + \widetilde{\mathbf{G}}(s)\widetilde{\mathbf{P}}(s). \tag{4.17}$$

After applying matrix algebra to Eqn. (4.17), the Laplace Stieltjes transform of the probability transition matrix $\tilde{P}(s)$ is given as

$$\widetilde{\mathbf{P}}(s) = (\mathbf{I} - \widetilde{\mathbf{G}}(s))^{-1}\widetilde{\mathbf{D}}(s). \tag{4.18}$$

By properties of Laplace and Laplace-Stieltjes transforms [30],

$$\mathbf{P}^*(s) = \frac{1}{s}\widetilde{\mathbf{P}}(s) = \frac{1}{s}(\mathbf{I} - \tilde{\mathbf{G}}(s))^{-1}\tilde{\mathbf{D}}(s). \tag{4.19}$$

Using a numerical approximation to the inverse Laplace transform yields a solution for the probability transition matrix $\mathbf{P}(t)$ in the time domain, where

$$\mathbf{P}(t) = \mathcal{L}^{-1}\left\{\frac{1}{s}(\mathbf{I} - \tilde{\mathbf{G}}(s))^{-1}\tilde{\mathbf{D}}(s)\right\}. \tag{4.20}$$

Using Eqn. (4.20) in conjunction with Eqns. (3.3) and (3.18), the satellite's instantaneous availability may be obtained.

The LST of $\mathbf{G}(t)$ does not exist in closed form for sojourn time distributions that are *power-tailed* and for some that are *heavy-tailed* [33]. A sojourn time distribution $H_i(t)$ is heavy-tailed if

$$\lim_{t\to\infty} \frac{\overline{H}_i(t+\gamma)}{\overline{H}_i(t)} = 1, \quad \gamma \geq 0 \tag{4.21}$$

and power-tailed if

$$\lim_{t\to\infty} \overline{H}_i(t) = t^{-\alpha}, \quad \alpha \geq 0, \quad t \geq 0. \tag{4.22}$$

For these types of c.d.f.s, a transform approximation method can be applied to approximate $\tilde{\mathbf{G}}(s)$ for Eqn. (4.20).

The following sections will show solution methods for obtaining the Laplace transform of the transition probability matrix $\mathbf{P}(t)$ for two-state SMPs. The solution of $\mathbf{P}^*(s)$ will be shown for two cases via calculation of $\widetilde{\mathbf{G}}(s)$ and $\widetilde{\mathbf{D}}(s)$. Each case will have a unique distribution for its state holding time. An approximation method from [18] is also presented for the Laplace transform of Weibull distributed state holding times, as the transform does not exist in closed form.

4.2.1 Example 1

Let the state holding time distributions, mean state holding estimates, and the associated estimated rates for each state of S be given as in Table 4.1. Let the

Table 4.1 Example 1 holding time distributions for a two-state SMP.

State	Distribution	Mean	r_i
1	Gamma(k, λ)	k/λ	λ/k
2	Uniform(a, b)	$(a+b)/2$	$2/(a+b)$

transition rate approximations from state i to state j be given as r_{ij}.

To calculate $\mathbf{P}^*(s)$, solutions for $\widetilde{\mathbf{G}}(s)$ and $\widetilde{\mathbf{D}}(s)$ are substituted into Eqn. (4.14) to obtain $\widetilde{\mathbf{P}}(s)$. Then by properties of Laplace and Laplace-Stieltjes transforms, division by s yields $\mathbf{P}^*(s)$. First,

$$G_{11}(t) = G_{22}(t) = 0,$$

by definition of the semi-Markov kernel, which implies that

$$\widetilde{G}_{11}(s) = \widetilde{G}_{22}(s) = 0.$$

For a two-state SMP,

$$\frac{r_{ij}}{r_i} = 1$$

since the only transition from state i is to state j and vice versa. Therefore,

$$G_{12}(t) \approx \frac{r_{12}}{r_1} H_1(t) = H_1(t), \ \text{ and } \ G_{21}(t) \approx \frac{r_{21}}{r_2} H_2(t) = H_2(t),$$

which implies that

$$\widetilde{G}_{12}(s) = \widetilde{H}_1(s), \ \text{ and } \ \widetilde{G}_{21}(s) = \widetilde{H}_2(s).$$

By definition of the matrix $\mathbf{D}(t)$,

$$D_{12}(t) = D_{21}(t) = 0, \ \ D_{11}(t) = 1 - H_1(t), \ \text{ and } \ D_{22}(t) = 1 - H_2(t).$$

The Laplace-Stieltjes transform has no effect on a constant. Therefore,

$$\widetilde{D}_{11}(s) = 1 - \widetilde{H}_1(s), \ \text{ and } \ \widetilde{D}_{22}(s) = 1 - \widetilde{H}_2(s).$$

From Eqn. (4.16) and from Table 4.1,

$$\widetilde{H}_1(s) = \left(\frac{\lambda}{\lambda + s}\right)^{\alpha} \ \text{ and } \ \widetilde{H}_2(s) = \frac{1}{s(b-a)}(e^{-sa} - e^{-sb}).$$

From these results, we obtain the matrices $\widetilde{\mathbf{G}}(s)$ and $\widetilde{\mathbf{D}}(s)$ as

$$\widetilde{\mathbf{G}}(s) = \begin{bmatrix} 0 & \left(\frac{\lambda}{\lambda+s}\right)^{\alpha} \\ \frac{1}{s(b-a)}(e^{-sa} - e^{-sb}) & 0 \end{bmatrix}$$

and

$$\widetilde{\mathbf{D}}(s) = \begin{bmatrix} 1-\left(\frac{\lambda}{\lambda+s}\right)^{\alpha} & 0 \\ 0 & 1-\frac{1}{s(b-a)}(e^{-sa}-e^{-sb}) \end{bmatrix}.$$

Therefore,

$$\begin{aligned}\mathbf{P}^*(s) &= \frac{1}{s}(\mathbf{I}-\tilde{\mathbf{G}}(s))^{-1}\tilde{\mathbf{D}}(s) \\ &= -\frac{\left(\frac{\lambda}{\lambda+s}\right)^{\alpha}(e^{-sa}-e^{-sb})}{s(b-a)} \begin{bmatrix} 1-\left(\frac{\lambda}{\lambda+s}\right)^{\alpha} & \left(\frac{\lambda}{\lambda+s}\right)^{\alpha}(1-\frac{(e^{-sa}-e^{-sb})}{s(b-a)}) \\ (1-\left(\frac{\lambda}{\lambda+s}\right)^{\alpha})\frac{(e^{-sa}-e^{-sb})}{s(b-a)} & 1-\frac{(e^{-sa}-e^{-sb})}{s(b-a)} \end{bmatrix}.\end{aligned}$$

Numerical methods can be used to perform the inverse Laplace transform to obtain $\mathbf{P}(t)$ in the time domain.

4.2.2 Example 2

Let the state holding time distributions, mean state holding estimates, and the associated estimated rates for each state of S be given as in Table 4.2. From

Table 4.2 Example 2 holding time distributions for a two-state SMP.

State	Distribution	Mean	r_i
1	Triangular(a, m, b)	$(a+m+b)/3$	$3/(a+m+b)$
2	Weibull(α, λ)	$\Gamma((\alpha+1)/\alpha)/\lambda$	$\lambda/\Gamma((\alpha+1)/\alpha)$

Example 1, it was shown that for a two-state SMP, $\mathbf{P}^*(s)$ is simply a function in terms of $\widetilde{H}_1(s)$ and $\widetilde{H}_2(s)$. From Eqn. (4.16) and Table 4.2,

$$\widetilde{H}_1(s) = \frac{2}{s^2}\left(\frac{e^{-sa}-e^{-sm}}{(b-a)(m-a)} + \frac{e^{-sb}-e^{-sm}}{(b-a)(b-m)}\right).$$

A closed form solution does not exist for $\widetilde{H}_2(s)$ due to the fact that the Weibull distribution is a heavy-tailed distribution and does not possess an analytical Laplace transform. Therefore, a numerical approximation to the Laplace transform must be used to obtain $\mathbf{P}^*(s)$.

The Transform Approximation Method (TAM) was developed for the purpose of finding an approximation to heavy-tailed distributions and is explained in numerous articles including [33] and [18]. A discrete approximation is performed on N discrete points in the domain of some c.d.f. $F(x)$, where each point is referred to as a TAM sample and is denoted $x(i)$, $i = 1, 2, \ldots, N$. Each of these points has an equal probability mass, resulting in a cumulative distribution estimate

$$F(x(i)) = \frac{i}{N}.$$

Then using a method from [22],

$$f^*(s) \approx \hat{f}^*(s) \equiv \frac{1}{N} \sum_{i=1}^{N} e^{-sx(i)}.$$

From [18], the TAM samples for the Weibull distribution are given by

$$x(i) = \beta \left[-\ln \left(1 - \frac{i}{N+1} \right) \right]^{\frac{1}{\alpha}},$$

resulting in the approximation

$$\widetilde{H}_2(s) \approx s\hat{H}^*(s) \equiv \frac{s}{N} \sum_{i=1}^{N} \exp \left[-s\beta \left[-\ln \left(1 - \frac{i}{N+1} \right) \right]^{\frac{1}{\alpha}} \right].$$

From $\widetilde{H}_1(s)$ and the approximation for $\widetilde{H}_2(s)$, $\mathbf{P}^*(s)$ can be obtained through algebra. Numerical methods can then be used to perform the inverse Laplace transform to obtain $\mathbf{P}(t)$ in the time domain.

4.3 Limiting Availability

The methods for obtaining the limiting availability for a satellite's degradation process modeled as a SMP are given below for comparison purposes. Let $A^{(k)}$ and

$A^{(k)}(t)$ denote the limiting availability and the instantaneous availability for satellite k at time t, respectively. Similar to Chapter 3, the limiting availability is solved by first obtaining the probability distribution for the degradation process in its asymptotic regime, then by summing up the long-run probabilities associated with states of availability.

Let $\mathbf{p}^{(k)}$ denote a row vector consisting of probabilities $p_i^{(k)}$ defined as

$$p_i^{(k)} = \lim_{t \to \infty} P(\mathbf{X}^{(k)}(t) = \sigma_i), \quad \sigma_i \in S, \quad i = 1, 2, \ldots, |S|.$$

The probability $p_i^{(k)}$ represents the long-run probability the satellite is found in state i. For a SMP that is irreducible and recurrent, the transition probability matrix $\mathbf{P} = \mathbf{G}(\infty)$. Let $\pi^{(k)}$ denote a row vector of probability values, where

$$\pi^{(k)} = \pi^{(k)}\mathbf{G}(\infty).$$

The system of equations

$$\begin{aligned} \pi^{(k)}\mathbf{P}^{(k)} &= \mathbf{0} \\ \pi^{(k)}\mathbf{e} &= \mathbf{1} \end{aligned}$$

is solved for $\pi^{(k)}$, where $\mathbf{P}^{(k)}$ is the transition probability matrix associated with satellite k, $\mathbf{0}$ denotes a row vector of zeros and $\mathbf{e}$ denotes a row vector of ones. As was stated earlier in the chapter, since the transition probabilities for the SMP satellite model are not known, we use statistics to approximate them. From Eqn. (4.10) and Eqn. (4.11), an approximation for $\pi^{(k)}$ can be solved from the system of equations

$$\begin{aligned} \pi^{(k)}\mathbf{R}^{(k)} &= \mathbf{0} \\ \pi^{(k)}\mathbf{e} &= \mathbf{1} \end{aligned}$$

where the matrix $\mathbf{R}^{(k)}$ represents the statistical estimator for the rate matrix of satellite k.

After solving for $\pi^{(k)}$, the probabilities of row vector $\mathbf{p}^{(k)}$ can be solved by the equation

$$p_i^{(k)} = \lim_{t \to \infty} P(\mathbf{X}^{(k)}(t) = i) = \frac{\pi_i^{(k)} \mu_i}{\sum_{i=1}^{|S|} \pi_i^{(k)} \mu_i},$$

where μ_i is the expected state holding time of state i. Finally, the limiting availability for satellite k is obtained by summing the long-term probabilities for the states of availability resulting in

$$A^{(k)} = \sum_{i=1}^{|S'|} p_i^{(k)}.$$

4.4 Model Summary

This chapter has presented a feasible extension to the formulation of the constellation functional capability measure. The satellite instantaneous availability measure was generalized by modeling the degradation process as a semi-Markov process. A methodology was described for acquiring the state probabilities at a given time when the holding time distributions are generally distributed. A brief overview of SMPs was provided along with examples of computing the Laplace transform of the probability transition matrix for a SMP with distinct sojourn time distributions. Finally, a methodology was described for obtaining the limiting availability of a satellite.

In the next chapter, we illustrate the methodologies presented in Chapters 3 and 4 with three numerical examples. Each example consists of calculating the constellation functional capability measure via the satellite instantaneous availability measures and value scores. Results from these examples are compared with those obtained via Monte Carlo simulation.

5. Numerical Results

This chapter presents a few examples to illustrate the methodology developed in Chapters 3 and 4. For each example, the sample space is described for the satellites contained within their parent constellation, then the instantaneous availability is calculated for each satellite based on the distribution of the failure and repair rates; the limiting availability is also calculated in the third example. Finally, each satellite is given a value score based on a set of defined attributes relating its contribution to the constellation's mission.

For these examples, the instantaneous availability and value scores are computed using notional data; however, the examples are carefully constructed to represent satellite design lifetimes from open literature resources. The results will demonstrate the benefits associated with the computational speed of using analytical methods rather than simulation models to calculate satellite availability as well as the errors prone to measuring a satellite's limiting availability over its instantaneous availability.

5.1 Description of Experiments

In this section we describe the means by which we compute the functional capability measures for each of the given examples. The comparison between the analytical and simulation methods is based on the computation of the satellite's instantaneous availability. The instantaneous availability measures for the analytical results were calculated using the CTMC methodology of Chapter 3 for example 1, and the SMP methodology of Chapter 4 for the remaining examples. Both the analytical results and the simulations were performed using MatLab software on a Dell Precision 670 computer with dual Intel Xeon 2.8 GHz processors with 3 GB of RAM. The satellite value scores and the final constellation functional capability measure were computed using Eqns. (3.28) and (3.1), respectively, of Chapter 3.

The first steps of computing the analytical instantaneous availability focused on constructing the rate matrix $\mathbf{Q}$, where a subprogram was written for this specific purpose. The subprogram first produces all of the state space elements, then determines which elements correspond to transitions of function failures and repairs. For the CTMC case, the rates were obtained from the failure and repair rates of the functions. For the SMP case, the distribution of the state holding times were used to construct the rates. By using the negative reciprocals of the expected state holding times for the diagonal entries, the remaining entries were constructed to make the rows sum to zero.

Next, the Laplace transform of the transition probability matrix $\mathbf{P}(t)$ was computed from Eqn. (4.19). Analytical expressions of the Laplace transforms of the distribution functions were used to calculate the transition probability matrix from the matrices $\tilde{\mathbf{G}}(s)$ and $\tilde{\mathbf{D}}(s)$. In the case where an analytical expression did not exist for the distribution, an approximation to the Laplace-Stieltjes transform was performed using the transform approximation method discussed in Chapter 4 with $N = 1000$. Numerical Laplace transform inversion was then used to obtain a time domain solution for the transition probability matrix. Each inverse Laplace transform was computed using the algorithm of Abate and Whitt [1]. Once the transition probability matrix was obtained, Eqn. (3.3) was solved to obtain the probability mass function, and Eqn. (3.18) was solved to obtain the satellite instantaneous availability.

A simulation model was also developed and executed to acquire the instantaneous availability results for each satellite in the constellation. These simulation results were compared to the analytical results in terms of processing time, as the actual numbers were indistinguishable in most cases. The simulation consisted of running multiple replications of the degradation process evolving randomly up to time t, the time at which the instantaneous availability is evaluated. The process began in state 1 and remained in this state for a random amount of time determined by a random number generator corresponding to the state holding distribution of

state 1. A subfunction randomly transitions the process to its next state by incorporating the rate matrix **Q**. The process remains in the next state for a random amount of time determined by a random number generator corresponding the that state's holding time distribution.

Transitions from one state to another were simulated with the same rate matrix used in the analytical computations. After each of the 0.5 million replications, a counter records the number of times the process is found in each of the possible states at time t. A probability mass function is constructed using these tallies. From this probability mass function, the states of availability are summed to obtain the simulated instantaneous availability of the satellite at time t.

The selected value scores for the three examples were based on the methodology presented in Chapter 3. Each example uses the same set of attributes to score the satellites. Values for constructing the attribute value functions, weights, and ultimately the satellite score, were notional. Using Eqn. (3.1), the constellation functional capability measure was computed with each satellite's instantaneous availability and value score.

5.2 Navstar GPS Constellation

The U.S. Military relies heavily on the Navstar Global Positioning System (GPS) to provide accurate time, location, and velocity data to mobile platforms such as aircraft and munitions [38]. The performance of this constellation depends on the capability of its 24 satellites to perform mission essential navigation functions. By calculating each satellite's instantaneous availability and scoring each satellite based on its operational contribution to the GPS constellation mission, the functional capability of this constellation will be assessed at $t = 9$ years.

Suppose that the constellation consists of 16 Block IIA satellites, and 8 Block IIR satellites, where the Block IIR satellites are newer, are designed with the latest

technology, and are built to last longer than the Block IIA satellites. Each satellite is independent of all other satellites and possesses three independent functions that have exponential failure and repair rates, the satellite's clock being the first function, the computer being the second, and the transceiver the third. A subject matter expert familiar with the constellation's mission and how its satellites contribute to the mission is used to determine a subset of the state space indicative of the satellite being available. For demonstrative purposes, suppose that in order for these satellites to be considered available, all three functions must be available.

5.2.1 Instantaneous Availability

The state space for each satellite will be identical and are defined in Table 5.1. Recall from Eqn. (3.2), the random variable indicating the state of a satellite is a vector of elements where a value of 1 refers to the function being available and a value of 0 refers to the function being unavailable. Figure 5.1 shows the transition rate

Table 5.1 Satellite states for the Navstar constellation example.

State	Combination	State	Combination
1	(1,1,1)	5	(0,0,1)
2	(0,1,1)	6	(0,1,0)
3	(1,0,1)	7	(1,0,0)
4	(1,1,0)	8	(0,0,0)

diagram for each satellite in this example, showing the possible transitions that can occur between states. The satellites are listed in Table 5.2 with their corresponding initial probability distributions and their years of initialization.

In this example, at the initial time satellites 17 through 20 start their useful life in the constellation, their status cannot be determined with certainty; there is a small probability associated with finding the satellite in state 3, corresponding to a 0.02 probability that the computer is not working at time t_0. This example illustrates how the model can incorporate a situation where the initial status of the satellite

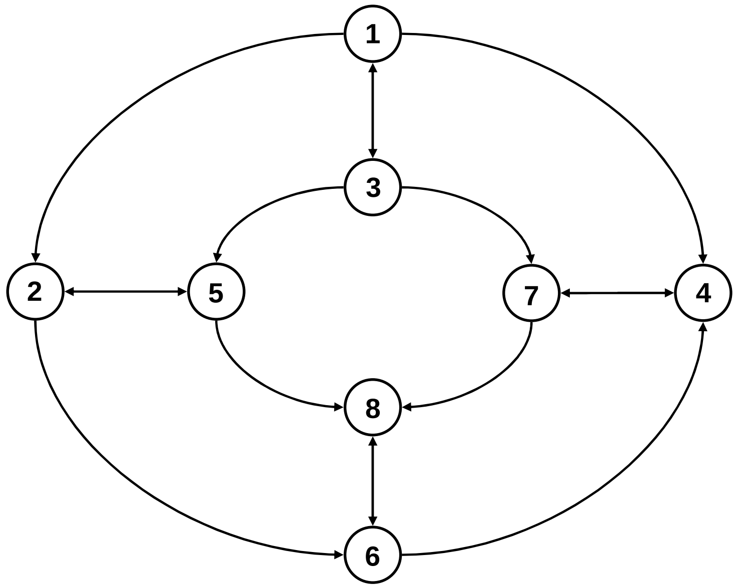

Figure 5.1 Transition rate diagram for the Navstar constellation example.

functions cannot be determined with certainty. Regardless of the interpretation of the satellite's initialization time t_0, the initial probability distribution contains the probabilities associated with the satellite being in a given state at t_0. If a satellite check-out can be performed at the time a satellite starts its life in the constellation, then the state of the satellite will be known with certainty.

Table 5.3 lists each function's failure and repair distribution. Notice that the computer is the only function considered to be repairable in this example.

Table 5.2 Satellite information for the Navstar constellation example.

Satellites	Type	Initial Probability Distribution	Time Initialized (t_0)
1-4	Block IIA	(1,0,0,0,0,0,0,0)	0
5-16	Block IIA	(1,0,0,0,0,0,0,0)	2
17-20	Block IIR	(0.98,0,0.02,0,0,0,0,0)	4
21-24	Block IIR	(1,0,0,0,0,0,0,0)	6

A graph of the instantaneous availability measures is given for each of the four groups of satellites in Figure 5.2, calculated using the methods of Chapter 3.

Table 5.3 Function information for the Navstar constellation example.

Satellites	Function	Failure Distribution	Repair Distribution
1-4	Clock	Exp(0.0291)	0
	Computer	Exp(0.0503)	Exp(3.3300)
	Transceiver	Exp(0.0230)	0
5-16	Clock	Exp(0.0121)	0
	Computer	Exp(0.0421)	Exp(4.8216)
	Transceiver	Exp(0.0189)	0
17-20	Clock	Exp(0.0121)	0
	Computer	Exp(0.0303)	Exp(4.9216)
	Transceiver	Exp(0.0143)	0
21-24	Clock	Exp(0.0103)	0
	Computer	Exp(0.0282)	Exp(4.9216)
	Transceiver	Exp(0.0118)	0

The simulated availability measures were not included in this figure as they were indistinguishable from the analytical measures. Notice that satellites 17 through 20 start with 0.98 availability at their initial times, corresponding to the uncertainty associated with the state these satellites may be in at time t_0. Table 5.4 summarizes the instantaneous availability measures for each of the satellites at $t = 9$ years. These measures take into account the amount of time the satellite has been in orbit by calculating its instantaneous availability based on its time of initialization, t_0. The average time for the analytical availability computation was 0.01 minutes versus 2.62 minutes for the simulation method. Thus, the computation increased by a factor of over 243 for the simulation results. The processing time for the simulation may be negligible in this case, however, the analytical solution may be a more feasible option for larger constellations containing satellites with more than three functions. Also, to obtain a simulation result with statistical confidence would require more replications, increasing the computational effort.

The limiting availability for all of the 24 satellites is equal to zero as some of the required functions cannot be repaired when they fail. When the satellite experiences a failure with its clock or with its transceiver, it is considered unavailable

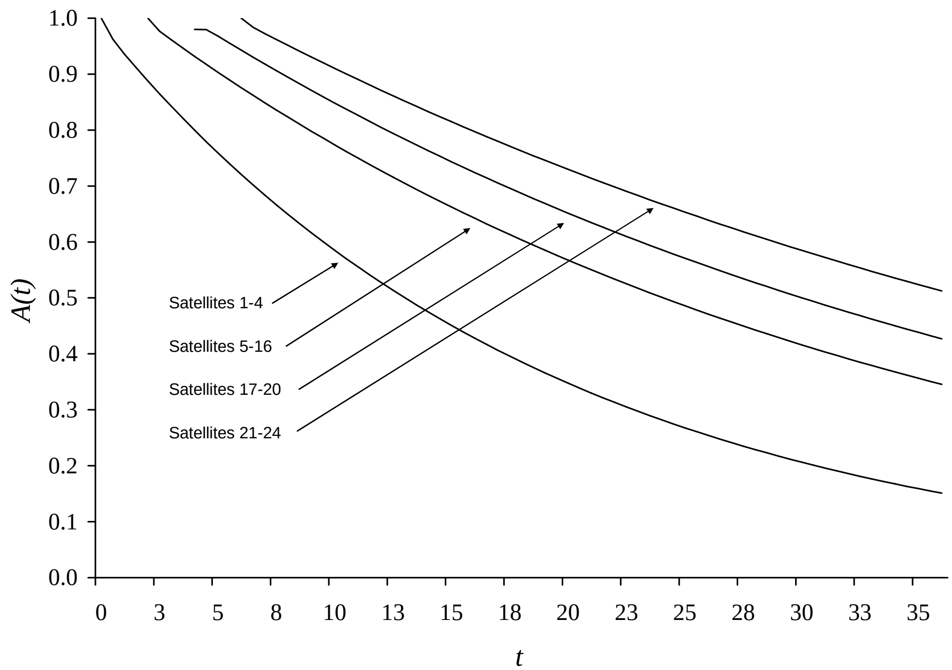

Figure 5.2 Comparison of instantaneous availability for Navstar satellites computed analytically.

and stays in this state for the remainder of its lifetime. Since the limiting availability is the proportion of uptime over total time, the numerator stays constant while the denominator approaches infinity, resulting in a limiting availability equal to zero.

5.2.2 Value Scores

Each satellite is assumed to be judged by a SME based on three independent attributes: (1) contribution to mission, (2) level of technology, and (3) signal accuracy. Attributes 1 and 2 were developed by [35]. These attributes are non-specific and could be applied to a number of different constellations. The normalized attribute scales are given in Table 5.5.

For each satellite, a value score is calculated using the methods of Chapter 3 by creating a value function and weight for each attribute, then having a SME score each attribute of each satellite. The value functions and attribute weights for the three attributes are given in Table 5.6 and are obtained from computing the risk

Table 5.4 Sample availability measures for the Navstar constellation example evaluated at $t = 9$ years.

			$A(t-t_0)$		
Satellites	t_0	$t-t_0$	Analytical	Simulated	Difference
1-4	0	9	0.616380	0.615982	0.000398
5-16	2	7	0.797963	0.797916	0.000047
17-20	4	5	0.870979	0.871156	0.000177
21-24	6	3	0.930518	0.930296	0.000222

Table 5.5 Normalized attribute scales for the Navstar constellation example.

Attribute	Measure (y)	Description
Contribution to Mission	0.00	No Direct Impact
	0.25	Secondary Impact to Mission
	0.50	Impacts Mission
	0.75	Strongly Impacts Mission
	1.00	Critical to Mission
Level of Technology	0.00	Primitive
	0.25	30 Year Old Technology
	0.50	20 Year Old Technology
	0.75	10 Year Old Technology
	1.00	State of the Art
Signal Accuracy	0.00	No Usable Signal
	0.33	Low Accuracy
	0.66	Medium Accuracy
	1.00	High Accuracy

attitude constants by solving Eqn. (3.30) with Newton's method. These scores are time dependent, as the attribute weights are assigned at the time of evaluation based on the preference of the attributes to the constellation's mission at time t. Table 5.7 summarizes the results associated with the value scores for each of the satellites. As in Chapter 3, $v_i(y_i)$ is the value function for attribute i evaluated at score y_i and $V^{(k)}(9)$ is the score for satellite k at time $t = 9$ years, calculated using Eqn. (3.28) with the satellite's value functions and the attribute weights.

Table 5.6 Value functions and attribute weights for the Navstar constellation example.

Attribute (i)	y_i'	c_i	$v_i(y_i)$	$w_i(9)$
1	0.60	0.822163	$\frac{1-\exp(0.822163 y_i)}{1-\exp(0.822163)}$	0.31
2	0.65	1.278652	$\frac{1-\exp(1.278652 y_i)}{1-\exp(1.278652)}$	0.29
3	0.66	1.376696	$\frac{1-\exp(1.376696 y_i)}{1-\exp(1.376696)}$	0.40

5.2.3 Functional Capability

Based on the instantaneous availability measurements along with the value scores for each satellite in the constellation, the constellation functional capability measure $\Phi(9)$ is calculated to be 0.710162 (0.710081 using simulation results).

5.3 Milstar Satellite Communication Constellation

The Milstar satellite communication system provides essential communication capabilities to the U.S. military through its five-satellite constellation [37]. The performance of this constellation is dependent on the capability of its 5 satellites to perform missions relating to secure, global communication. By calculating each satellite's instantaneous availability and scoring each satellite based on its operational contribution to the Milstar constellation mission, the functional capability of this constellation will be assessed at $t = 9$ years using the methodology described in Chapters 3 and 4.

For this example, suppose that the constellation consists of 5 identical and independent satellites. Each satellite consists of an independent function for each of the four types of information it processes: voice, data, teletype, and facsimile. Further suppose that in order for these satellites to be considered available, the first two functions must be available; assume that if the teletype and facsimile functions become unavailable, the availability of the data function keeps the satellite in an

Table 5.7 Value score results for the Navstar constellation example.

Satellite	y_1	y_2	y_3	$v_1(y_1)$	$v_2(y_2)$	$v_3(y_3)$	$V(9)$
1	0.84	0.79	0.98	0.780095	0.673654	0.963672	0.822658
2	0.86	0.79	0.94	0.806027	0.673654	0.893949	0.802808
3	0.84	0.79	0.86	0.780095	0.673654	0.765511	0.743393
4	0.84	0.79	0.95	0.780095	0.673654	0.911022	0.801598
5	0.96	0.83	0.97	0.942283	0.729252	0.945880	0.881943
6	0.86	0.83	0.98	0.806027	0.729252	0.963672	0.846820
7	0.95	0.83	0.94	0.928148	0.729252	0.893949	0.856789
8	0.97	0.83	0.99	0.956535	0.729252	0.981711	0.900693
9	0.98	0.83	0.99	0.970904	0.729252	0.981711	0.905148
10	0.94	0.83	0.98	0.914128	0.729252	0.963672	0.880332
11	0.86	0.83	0.99	0.806027	0.729252	0.981711	0.854036
12	0.95	0.83	0.89	0.928148	0.729252	0.812026	0.824019
13	0.83	0.83	0.96	0.767288	0.729252	0.928331	0.820675
14	0.94	0.83	0.98	0.914128	0.729252	0.963672	0.880332
15	0.86	0.83	0.99	0.806027	0.729252	0.981711	0.854036
16	0.95	0.83	0.96	0.928148	0.729252	0.928331	0.870541
17	0.83	0.94	0.99	0.767288	0.897656	0.981711	0.890864
18	0.99	0.94	0.99	0.985392	0.897656	0.981711	0.958476
19	0.89	0.94	0.98	0.845734	0.897656	0.963672	0.907967
20	0.96	0.94	0.97	0.942283	0.897656	0.945880	0.930780
21	0.98	0.97	0.99	0.970904	0.947847	0.981711	0.968540
22	0.97	0.97	0.98	0.956535	0.947847	0.963672	0.956870
23	0.99	0.97	0.99	0.985392	0.947847	0.981711	0.973032
24	0.98	0.97	0.99	0.970904	0.947847	0.981711	0.968540

available status. Also, suppose that the only function that can be repaired on the satellite is the voice function, provided that the data function is still available.

5.3.1 Instantaneous Availability

The state space for each satellite will be identical and is defined in Table 5.8. Notice that states 1, 4, 5, and 11 are states of availability for this satellite, corresponding to having both the voice and data functions available. Figure 5.3 shows the possible transitions that can occur between states. The satellites are

Table 5.8 Satellite states for the Milstar constellation example.

State	Combination	State	Combination
1	**(1,1,1,1)**	9	(1,0,0,1)
2	(0,1,1,1)	10	(1,0,1,0)
3	(1,0,1,1)	**11**	**(1,1,0,0)**
4	**(1,1,0,1)**	12	(0,0,0,1)
5	**(1,1,1,0)**	13	(0,0,1,0)
6	(0,0,1,1)	14	(0,1,0,0)
7	(0,1,0,1)	15	(1,0,0,0)
8	(0,1,1,0)	16	(0,0,0,0)

listed in Table 5.9 with their corresponding initial probability distribution, and their year of initialization.

Table 5.10 lists the state holding time distributions for each of the states for this example.

Table 5.9 Satellite information for the Milstar constellation example.

Satellite	Initial Probability Distribution	Time Initialized (t_0)
1	(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)	0
2	(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)	4
3	(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)	6
4	(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)	7
5	(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)	8

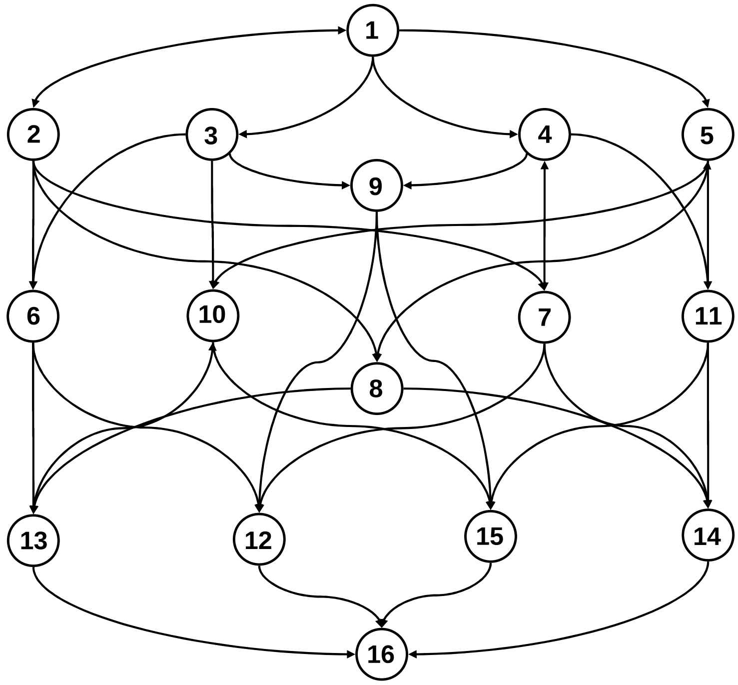

Figure 5.3 Transition rate diagram for the Milstar constellation example.

Let the statistical estimates for the transition rates be given in the matrix $\mathbf{R}$, where

$$\mathbf{R} = \begin{bmatrix}
-0.097 & 0.019 & 0.029 & 0.029 & 0.019 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.048 & -0.238 & 0.000 & 0.000 & 0.000 & 0.095 & 0.048 & 0.048 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.000 & -0.186 & 0.000 & 0.000 & 0.037 & 0.000 & 0.000 & 0.075 & 0.075 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.000 & -0.131 & 0.000 & 0.000 & 0.066 & 0.000 & 0.039 & 0.000 & 0.026 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & -0.234 & 0.000 & 0.000 & 0.047 & 0.000 & 0.164 & 0.023 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.189 & 0.000 & 0.000 & -0.472 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.142 & 0.142 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.175 & 0.000 & 0.000 & -0.349 & 0.000 & 0.000 & 0.000 & 0.000 & 0.070 & 0.000 & 0.105 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.140 & 0.000 & 0.000 & -0.351 & 0.000 & 0.000 & 0.000 & 0.000 & 0.175 & 0.035 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & -0.484 & 0.000 & 0.000 & 0.339 & 0.000 & 0.000 & 0.145 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & -0.428 & 0.000 & 0.000 & 0.342 & 0.000 & 0.086 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & -0.563 & 0.000 & 0.000 & 0.338 & 0.225 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.069 & 0.000 & 0.000 & -0.344 & 0.000 & 0.000 & 0.000 & 0.276 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.255 & 0.000 & 0.000 & -0.851 & 0.000 & 0.000 & 0.596 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.269 & 0.000 & 0.000 & -0.897 & 0.000 & 0.628 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & -0.823 & 0.823 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000
\end{bmatrix}.$$

Table 5.10 State holding time distributions for the Milstar constellation example.

State	Distribution	State	Distribution
1	Gamma(3.80,0.37)	9	Triangular(1.80,2.10,2.30)
2	Gamma(2.10,0.50)	10	Triangular(1.61,2.20,3.20)
3	Gamma(2.20,0.41)	11	Triangular(1.10,1.83,2.40)
4	Gamma(3.20,0.42)	12	Triangular(1.70,2.50,4.51)
5	Weibull(1.20,0.22)	13	Uniform(0.05,2.30)
6	Weibull(1.80,0.42)	14	Uniform(0.13,2.10)
7	Weibull(1.90,0.31)	15	Uniform(0.03,2.40)
8	Weibull(1.20,0.33)	16	Uniform(0.51,0.93)

Since the voice function is the only function that can be repaired, the approximation **R** to the rate matrix **Q** is sparse below its diagonal. (A satellite consisting of functions that cannot be repaired results in a lower triangular rate matrix.)

A graph of the instantaneous availability compared to the simulated availability for the satellites in this constellation is given in Figure 5.4, where the instantaneous availability is calculated using Eqns. (4.6)-(4.20) of Chapter 4, and Eqns. (3.3) and (3.18) of Chapter 3. Note that the instantaneous availability measures for each of the 5 satellites can be derived from the same analysis by accounting for the year of initialization. Table 5.11 summarizes the instantaneous availability measures for each of the satellites at $t = 9$ years. The average time for the analytical availability computation was 1.16 minutes versus 10.47 minutes for the simulation method. Thus, the computation increased by a factor of over 9 for the simulation results.

Table 5.11 Sample availability measures for the Milstar constellation example evaluated at $t = 9$ years.

			$A(t - t_0)$		
Satellite	t_0	$t - t_0$	Analytical	Simulated	Difference
1	0	9	0.722399	0.740352	0.017953
2	4	5	0.903839	0.912136	0.008297
3	6	3	0.961779	0.965656	0.003877
4	7	2	0.980183	0.981590	0.001407
5	8	1	0.991859	0.992288	0.000429

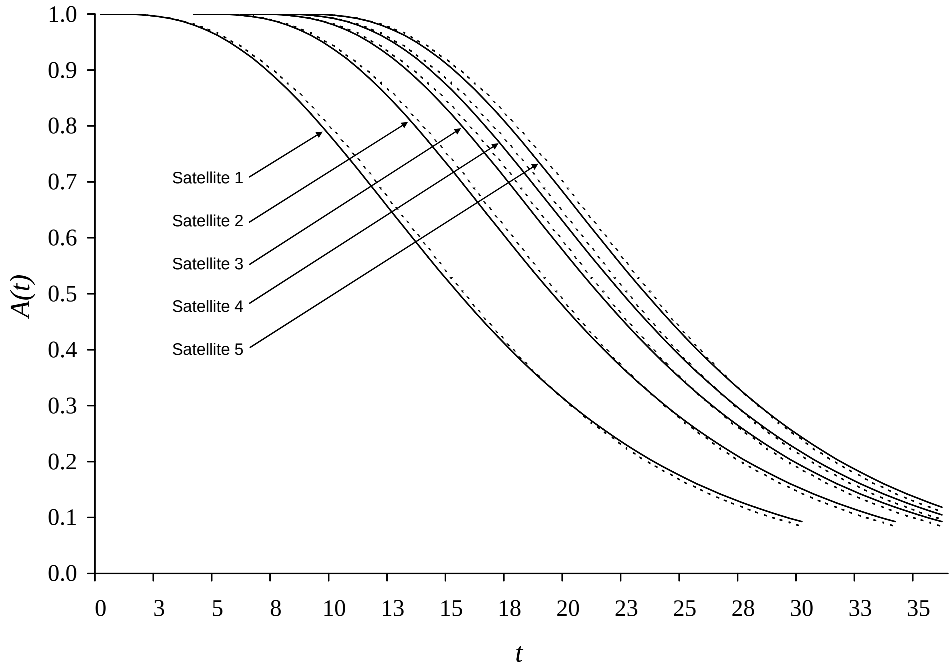

Figure 5.4 Comparison of instantaneous availability for the Milstar satellites computed analytically (solid line) and simulated availability (dotted line).

As was the case with the Navstar constellation example, the limiting availability for all of the Milstar constellation satellites is equal to zero as there exist functions that cannot be repaired which are necessary for the satellite to be considered available. For example, once the satellite reaches state 16, it will never leave state 16 as there are no repairs that will take place. Since the data function is required for the satellite to be classified as available, and the data function cannot be repaired, the satellite's availability is limited by the life of the data function.

5.3.2 Value Scores

The value scores of the Milstar constellation will be calculated using the same set of attributes that were used to score the satellites in the Navstar constellation example. The value functions and attribute weights for the three attributes are given

in Table 5.19, and Table 5.20 summarizes the results associated with the value scores for each of the satellites.

Table 5.12 Value functions and attribute weights for the Milstar constellation example.

Attribute (i)	y_i'	c_i	$v_i(y_i)$	$w_i(9)$
1	0.73	2.164803	$\frac{1-\exp(2.164803y_i)}{1-\exp(2.164803)}$	0.57
2	0.64	1.183190	$\frac{1-\exp(1.183190y_i)}{1-\exp(1.183190)}$	0.29
3	0.55	0.402692	$\frac{1-\exp(0.402692y_i)}{1-\exp(0.402692)}$	0.14

Table 5.13 Value score results for the Milstar constellation example.

Satellite	y_1	y_2	y_3	$v_1(y_1)$	$v_2(y_2)$	$v_3(y_3)$	$V(9)$
1	0.94	0.73	0.81	0.862401	0.605795	0.777793	0.776140
2	0.96	0.83	0.94	0.906296	0.737345	0.927984	0.860337
3	0.93	0.90	0.86	0.841156	0.839142	0.834630	0.839658
4	0.98	0.94	0.95	0.952134	0.901211	0.939866	0.935649
5	0.96	0.96	0.97	0.906296	0.933364	0.963775	0.922193

5.3.3 Functional Capability

Based on the instantaneous availability measurements along with the value scores for each satellite in the constellation, the constellation functional capability measure $\Phi(9)$ is calculated to be 0.795529 (0.800737 using simulation results).

5.4 Defense Meteorological Satellite Constellation

The U.S. Air Force collects weather data from a constellation of two meteorological satellites in polar orbits [36]. These satellites have multiple functions associated with performing their mission. The satellite's primary system collects visual and infrared imagery data. There are also secondary sensors on the satellite, one which "measure[s] atmospheric vertical profiles of moisture and temperature" and

another which measures the ionosphere to determine its impact on other space-based systems [36:1]. We will consider the satellites in this constellation to be independent and satellite 1 has four functions defined in Table 5.14 and satellite 2 has only the first three of these functions. Each satellite will be considered available if all of its functions are available. Also, all of the functions will be considered repairable. The instantaneous availability and value score will be calculated for both satellites of this constellation to compute the constellation functional capability measure at $t = 10$ years.

Table 5.14 Satellite functions for the meteorological constellation example.

Function	Description
1	Visual imagery sensor
2	Infrared imagery sensor
3	Vertical moisture and temperature profile sensor
4	Ionosphere sensor

5.4.1 Instantaneous Availability

The state space for satellite 1 is the same as the state space for the satellites in the Milstar constellation example (Table 5.8). The state space for satellite 2 is the same as the state space for the satellites in the Navstar constellation example (Table 5.1). The transition rate diagram for satellite 1 is similar to the diagram in Figure 5.3 for the Milstar constellation example. The transition rate diagram for satellite 2 is similar to the diagram in Figure 5.1 for the Navstar constellation example. Since all of the functions can be repaired on the meteorological constellations satellites, all of the states communicate, resulting in transition rate diagrams with two-direction arcs.

Both of these satellites are assumed to have initial probability distributions where the probability of being in state 1 is equal to 1.0. Both satellites will also be assumed to have initialized at the same time. Tables 5.15 and 5.16 list the state

holding time distributions for each of the states associated with satellite 1 and 2, respectively.

Table 5.15 State holding time distributions for satellite 1 of the meteorological constellation example.

State	Distribution	State	Distribution
1	Gamma(5.80,0.37)	9	Uniform(0.05,1.40)
2	Gamma(4.10,6.80)	10	Uniform(0.13,1.10)
3	Gamma(1.60,2.31)	11	Triangular(0.02,0.50,1.30)
4	Gamma(1.20,1.13)	12	Triangular(0.13,0.40,1.20)
5	Erlang(2,7.41)	13	Triangular(0.09,0.30,1.10)
6	Erlang(5,6.42)	14	Exp(1.60)
7	Weibull(1.20,1.22)	15	Exp(1.40)
8	Weibull(1.80,2.20)	16	Exp(1.80)

Table 5.16 State holding time distributions for satellite 2 of the meteorological constellation example.

State	Distribution	State	Distribution
1	Gamma(8.70,1.07)	5	Weibull(1.20,1.22)
2	Gamma(3.10,5.80)	6	Weibull(1.80,2.20)
3	Erlang(2,6.41)	7	Triangular(0.02,0.50,1.30)
4	Erlang(5,6.42)	8	Exp(1.80)

Let the statistical estimates for the transition rates for satellite 1 and 2 be given in the matrices $\mathbf{R}^{(1)}$ and $\mathbf{R}^{(2)}$, respectively, where

$$\mathbf{R}^{(1)} = \begin{bmatrix}
-0.064 & 0.013 & 0.019 & 0.019 & 0.013 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.663 & -1.659 & 0.000 & 0.000 & 0.000 & 0.332 & 0.332 & 0.332 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.722 & 0.000 & -1.444 & 0.000 & 0.000 & 0.144 & 0.000 & 0.000 & 0.289 & 0.289 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.188 & 0.000 & 0.000 & -0.942 & 0.000 & 0.000 & 0.377 & 0.000 & 0.188 & 0.000 & 0.188 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
1.482 & 0.000 & 0.000 & 0.000 & -3.705 & 0.000 & 0.000 & 0.371 & 0.000 & 1.482 & 0.371 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.385 & 0.385 & 0.000 & 0.000 & -1.284 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.128 & 0.385 & 0.000 & 0.000 & 0.000 \\
0.000 & 0.259 & 0.000 & 0.519 & 0.000 & 0.000 & -1.297 & 0.000 & 0.000 & 0.000 & 0.000 & 0.259 & 0.000 & 0.259 & 0.000 & 0.000 \\
0.000 & 0.742 & 0.000 & 0.000 & 0.742 & 0.000 & 0.000 & -2.474 & 0.000 & 0.000 & 0.000 & 0.000 & 0.742 & 0.247 & 0.000 & 0.000 \\
0.000 & 0.000 & 0.414 & 0.414 & 0.000 & 0.000 & 0.000 & 0.000 & -1.379 & 0.000 & 0.000 & 0.414 & 0.000 & 0.000 & 0.138 & 0.000 \\
0.000 & 0.000 & 0.325 & 0.000 & 0.488 & 0.000 & 0.000 & 0.000 & 0.000 & -1.626 & 0.000 & 0.000 & 0.488 & 0.000 & 0.325 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.495 & 0.165 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & -1.648 & 0.000 & 0.000 & 0.495 & 0.495 & 0.000 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.347 & 0.520 & 0.000 & 0.347 & 0.000 & 0.000 & -1.734 & 0.000 & 0.000 & 0.000 & 0.520 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.604 & 0.000 & 0.604 & 0.000 & 0.201 & 0.000 & 0.000 & -2.013 & 0.000 & 0.000 & 0.604 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.320 & 0.640 & 0.000 & 0.000 & 0.480 & 0.000 & 0.000 & -1.600 & 0.000 & 0.160 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.280 & 0.420 & 0.420 & 0.000 & 0.000 & 0.000 & -1.400 & 0.280 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.000 & 0.720 & 0.720 & 0.180 & 0.180 & -1.800
\end{bmatrix}$$

and

$$\mathbf{R}^{(2)} = \begin{bmatrix}
-0.123 & 0.061 & 0.037 & 0.025 & 0.000 & 0.000 & 0.000 & 0.000 \\
0.748 & -1.871 & 0.000 & 0.000 & 0.561 & 0.561 & 0.000 & 0.000 \\
0.962 & 0.000 & -3.205 & 0.000 & 0.962 & 0.000 & 1.282 & 0.000 \\
0.385 & 0.000 & 0.000 & -1.284 & 0.000 & 0.642 & 0.257 & 0.000 \\
0.000 & 0.648 & 0.519 & 0.000 & -1.297 & 0.000 & 0.000 & 0.130 \\
0.000 & 0.990 & 0.000 & 0.742 & 0.000 & -2.474 & 0.000 & 0.742 \\
0.000 & 0.000 & 0.659 & 0.495 & 0.000 & 0.000 & -1.648 & 0.495 \\
0.000 & 0.000 & 0.000 & 0.000 & 0.540 & 0.540 & 0.720 & -1.800
\end{bmatrix}.$$

Figure 5.5 shows the instantaneous availability and limiting availability for both of the satellites in the meteorological constellation. The simulated availability measures were not included in this figure as they were indistinguishable from the analytical measures. Table 5.17 summarizes the instantaneous availability measures for each of the satellites at $t = 10$ years. The average time for the analytical availability computation was 0.37 minutes versus 13.03 minutes for the simulation method. Thus, the computation increased by a factor of over 34 for the simulation results.

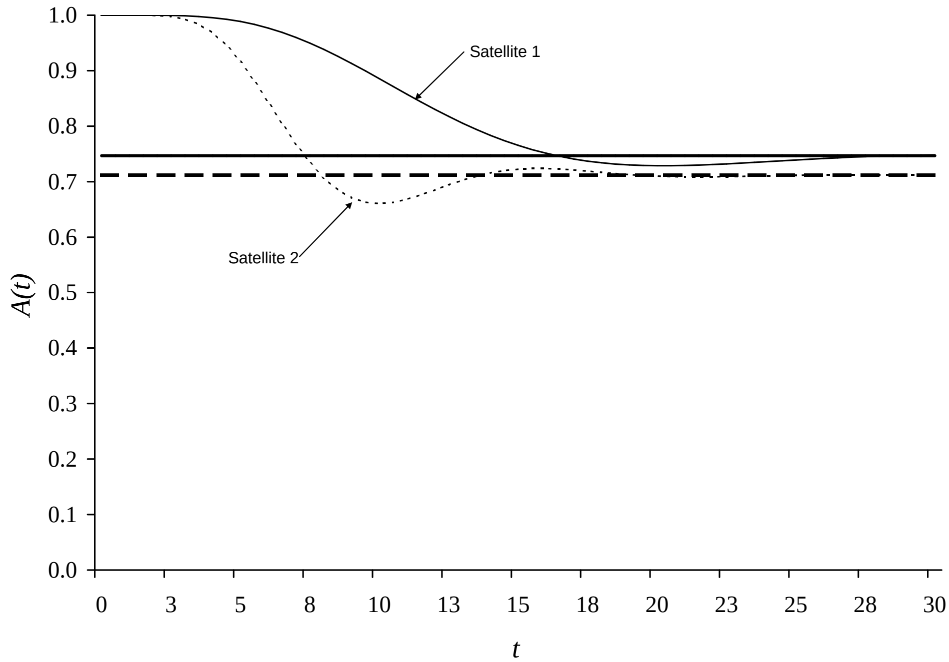

Figure 5.5 Comparison of instantaneous availability (regular) with the limiting availability (bold) for satellite 1 (regular) and satellite 2 (dotted) of the meteorological constellation example.

Table 5.17 Sample availability measures for the meteorological constellation example evaluated at $t = 10$ years.

		$A(t)$		
Satellite	t	Analytical	Simulated	Difference
1	10	0.885548	0.885096	0.000452
2	10	0.660505	0.660400	0.000105

The limiting availability is calculated for satellites 1 and 2 using the methods described in Chapter 4. Table 5.18 shows the differences between the instantaneous availability and the limiting availability. Notice that the limiting availability measure for satellite 1 underestimates the true availability of the satellite by 13%, whereas the limiting availability measure for satellite 2 overestimates the true availability of the satellite by 5%.

Table 5.18 Comparison of instantaneous and limiting availability measures for the meteorological constellation example.

Satellite	t	Instantaneous Availability	Limiting Availability	Difference
1	10	0.885548	0.746574	0.138974
2	10	0.660505	0.711398	0.050893

5.4.2 Value Scores

The value scores of the meteorological constellation will be calculated using the same set of attributes that were used to score the satellites of the previous constellation examples. The value functions and attribute weights for the three attributes are given in Table 5.19, and Table 5.20 summarizes the results associated with the value scores for each of the satellites.

Table 5.19 Value functions and attribute weights for the meteorological constellation example.

Attribute (i)	y_i'	c_i	$v_i(y_i)$	$w_i(10)$
1	0.70	1.801072	$\frac{1-\exp(1.801072 y_i)}{1-\exp(1.801072)}$	0.50
2	0.65	1.278652	$\frac{1-\exp(1.278652 y_i)}{1-\exp(1.278652)}$	0.32
3	0.45	-0.402692	$\frac{1-\exp(-0.402692 y_i)}{1-\exp(-0.402692)}$	0.18

Table 5.20 Value score results for the meteorological constellation example.

Satellite	y_1	y_2	y_3	$v_1(y_1)$	$v_2(y_2)$	$v_3(y_3)$	$V(10)$
1	0.99	0.77	0.86	0.978620	0.646903	0.883036	0.855265
2	0.97	0.77	0.89	0.936999	0.646903	0.908657	0.839067

5.4.3 Functional Capability

Based on the instantaneous availability measurements along with the value scores for each satellite in the constellation, the constellation functional capability measure $\Phi(10)$ is calculated to be 0.655793 (0.655556 using simulation results).

Using the limiting availability measures to calculate the constellation functional capability measure yields 0.617714, which underestimates the constellations functional capability by 3.8%. An inaccurate constellation functional capability measure could lead to the unnecessary repair or replacement of satellites in the constellation in the attempt to maintain the constellation's level of capability.

5.5 Discussion

This chapter illustrated the techniques developed in Chapters 3 and 4 in three different constellations. For each of the examples, the satellite instantaneous availability and value score were calculated to generate the constellation's functional capability measure. The availability measures were computed using analytical methods and Monte Carlo simulation, yielding indistinguishable results. This indicates that errors associated with the numerical approximations used in the analytical method, such as transform approximations and numerical inversions, are insignificant when calculating the constellation's functional capability. Moreover, the analytical results were computed with substantially less effort (i.e., reduced computational time) than those generated via Monte Carlo simulation. A summary of the processing times for each example is presented in Table 5.21. The benefits associated with the shorter processing time of the analytical method would be realized for constellations modeled with more functions, equating to a larger state space.

Table 5.21 Comparison of average processing time (mins) for instantaneous availability measures for the constellation examples.

Example	Analytical	Simulation	Factor Increase
Navstar Constellation	0.01	2.62	243.6
Milstar Constellation	1.16	10.47	9.1
Meteorological Constellation	0.37	13.03	34.8

As stated in Chapter 1, the purpose of the constellation functional capability measure is to provide insight into the capability of the constellation in terms of

mission effectiveness. A replenishment policy might base the decision to perform a satellite repair or launch a new satellite based on a minimum threshold of constellation functional capability. A Type-I error associated with the constellation capability measure would imply that the true constellation capability is higher than what the measure reports it to be. This might result in unnecessary satellite repairs or replacements, thus increasing the overall cost of maintaining the satellite constellation. A Type-II error would imply that the true constellation capability is lower than what the measure reports it to be. This might result in the failure to maintain the constellation above the minimum threshold of functional capability, possibly jeopardizing missions which rely on the services provided by the constellation.

In the first two examples, the limiting availability of the satellites is zero since their availability depends on functions that cannot be repaired once they have failed. The limiting availability computed for the third example highlighted errors that can occur as a result of using a limiting availability in place of instantaneous availability. Had the constellation's functional capability been assessed at an earlier time, these errors would have been even larger. The next chapter will offer conclusions and recommendations to this research as well as suggestions for further research.

6. Conclusions and Future Research

This thesis has proposed a new functional capability measure for a satellite constellation. The methodology is consistent with the definitions of performance measures found in the literature, incorporating the constellation's availability and effectiveness. Chapter 1 introduced constellation capability measures and discussed their alignment with capability-based assessment of U.S. military systems. This measure is a necessary component for capability-based replenishment policies, possibly giving space system planners information to justify program funds. Chapter 2 showed how this research extends the literature; a constellation specific measure is developed that incorporates a stochastic model of satellite availability with multi-attribute value theory.

Chapters 3 and 4 respectively presented mathematical models for the explicit computation of the functional capability measure. Each satellite's instantaneous availability and value score are calculated at a common constellation age yielding a functional capability score. The constellation measure is formed by averaging these capability scores. By employing a few reasonable assumptions, the satellite degradation is modeled as a stochastic process to acquire its instantaneous availability. Finally, Chapter 5 illustrated the application of the methodology in three constellation examples. The functional capability measures were computed for a constellation via individual satellite availability and value scores. Results of the instantaneous availability computation were compared with those obtained by Monte Carlo simulation and were shown to be consistent.

The methodology proposed in this thesis results in a mathematically tractable solution, allowing the measure to be directly computed. The speed of computation for the instantaneous availability measures was considerably faster than that required for the Monte Carlo simulation results. The errors associated with the numerical inversion and transform approximation methods of the analytical solu-

tion were negligible, yielding an indistinguishable constellation capability measure. Incorporating the instantaneous availability model to a satellite with M functions, does require computation of matrices of dimension 2^M. Satellites with many functions resulting in large state spaces could possibly lead to computational complexity issues.

The imposed assumptions for the stochastic degradation process of the satellite are not entirely unrealistic. The model does not require satellites to be identical. The meteorological constellation example presented in Chapter 5 demonstrated this flexibility. Also, in the case of a satellite with exponential function lifetimes, the methodology of Chapter 3 is exact. The methodology of Chapter 4 to a satellite with non-exponential function lifetimes assumes the sojourn time distributions are independent of the next transition state. This may be an unrealistic assumption given the manor in which the satellite's state space is constructed. However, a model requiring all of the conditional sojourn time distributions between all possible states would be infeasable, as acquiring data to parametrically estimate these distributions is nearly impossible.

The independence assumption for functions and satellites may also be unrealistic. The main subsystems of a satellite are designed with as much redundancy as possible to prevent catastrophic satellite failure in the event that one function fails. In reality, the satellite payload depends on the satellite bus, which provides a platform for the mission related functions. In terms of satellite independence, some constellations have designated relay satellites, which act as the communication link from ground stations to the constellation. Other satellites within the constellation are dependent on this relay satellite to perform their functions. Therefore, this model assumption may be violated for certain constellations.

The methodology is not specific to one type of constellation mission. The satellite value scores are based on attributes constructed to directly reflect mission effectiveness of the constellation. However, the value scores may be entirely sub-

jective. These scores depend on accurate information from a knowledgable subject matter expert to construct and score each of the satellites in the constellation. The value score is ultimately the opinion of a SME; constellation measures may not be consistent when judged with different SMEs. Moreover, there are many methods for generating a multi-attribute value; this thesis presented only one of them. Certain methods may be more advantageous to specific satellite constellations, and also to particular subject matter experts.

Our proposed model for constellation functional capability can benefit from improvements which generalize and facilitate its application so we now discuss a few suggestions for model improvements. First, this model only encompasses the space segment of an entire system that accomplishes space-based missions. The model can be expanded to include the ground segment of the system as well. Second, the idea of on-orbit satellite spares could be incorporated into the model. A possible way to accomplish this might involve constructing the state space in a manner which allows multiple failed functions to be brought back to an available state simultaneously, simulating the entire replacement of a satellite.

The method for constructing the satellite value scores is another area that can be improved. In particular, the value score may be incorporated as a reward within a Markov reward model that can be analyzed using a number of existing techniques. Finally, a different approach to modeling satellites with non-exponential function lifetimes is to use phase-type distributions, allowing non-exponential distributions to be transformed into exponential distributions.

Appendix A. Computer Code

A.1 Analytical Availability

```
function [t,A_SMP,A_stat,duration] = Availability_Example1(t)
% Author:       Capt Cole Gulyas
% Date:         19 Jan 05
% Input:        t=time
% Output:       Measures for Example 1: t=time, A_SMP=instantaneous availability,
%               A_stat=stationary availability, and duration=processing time.
% Subfunctions: prob_matrix(Q,t) by Capt Cole Gulyas, Dr. Jeffrey Kharoufeh
%               invt_lap(Q,t,row,col) by Dr. Jeffrey Kharoufeh
%               e135a(Q,x,y,row,col) by Dr. Jeffrey Kharoufeh, Capt Cole Gulyas
%               rate_matrix(m,f,r) by Capt Cole Gulyas

tic                 % Start timer

%1-4 Block A
m=3;
n=2^m;
a=[1 0 0 0 0 0 0 0];
s=[1 0 0 0 0 0 0 0];
f=[.0291 .0503 .0230];
r=[0 3.3300 0];

% %5-16 Block A
% m=3;
% n=2^m;
% a=[1 0 0 0 0 0 0 0];
% s=[1 0 0 0 0 0 0 0];
% f=[.0121 .0421 .0189];
% r=[0 4.8216 0];

% %17-20 Block R
% m=3;
% n=2^m;
% a=[.98 0 .02 0 0 0 0 0];
% s=[1 0 0 0 0 0 0 0];
% f=[.0121 .0303 .0143];
% r=[0 4.9216 0];

% %21-24 Block R
```

```
% m=3;
% n=2^m;
% a=[1 0 0 0 0 0 0 0];
% s=[1 0 0 0 0 0 0 0];
% f=[.0103 .0282 .0118];
% r=[0 4.9216 0];

Q=rate_matrix(m,f,r);

% Mean holding times for each state
mu=-1./diag(Q);

% P*(s)
P_Laplace_SMP=prob_matrix(Q,t);

% Pmf
pmf_SMP=a*P_Laplace_SMP;

% Availability
A_SMP=pmf_SMP*s';

duration=toc;       % Stop timer

% Calculate stationary distribution and availability
% Pi
ze=zeros(1,n);
ze(end)=1;
Pe=[];
for i=1:n
    for j=1:n
        if i~=j Pe(i,j)=-Q(i,j)/Q(i,i); end
    end
end
Pe=Pe-eye(n);
Pe(:,end)=ones(n,1);
pie=ze*inv(Pe);

% Pmf and stationary availability
p=[];
for j=1:n
    p=[p (pie(j)*mu(j))/(pie*(mu))];
end
A_stat=p*s';
```

```
%-----------Sub Function-----------------------------------------------
function eq = e135a(Q,x,y,row,col)
% Author:           Capt Gulyas, Dr. Jeffrey Kharoufeh
% Last Revision:    4 Oct 04
% Note: function handles Q, row, and col were added to function e135a

s=x+y*i;                               % Complex transform variable
P_star = [];                           % Declare matrix
I=eye(size(Q));                        % Identity matrix

% All states are distributed exponential:
LSTG=[];
LSTD=[];
for a=1:length(Q)
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*(-Q(a,a)/(s-Q(a,a))); end
        LSTD(a,a)=s/(s-Q(a,a));
    end
end

P_star = (1/s)*(inv(I-LSTG))*LSTD;       % Build P*(s)
eq = real(P_star(row,col));              % Return first moment
%-----------End Sub Function-------------------------------------------


function [t,A_SMP,A_stat,duration] = Availability_Example2(t)
% Author:       Capt Cole Gulyas
% Date:         5 Jan 05
% Input:        t=time
% Output:       Measures for Example 2: t=time, A_SMP=instantaneous availability,
%               A_stat=stationary availability, and duration=processing time.
% Subfunctions: prob_matrix(Q,t) by Capt Cole Gulyas, Dr. Jeffrey Kharoufeh
%               invt_lap(Q,t,row,col) by Dr. Jeffrey Kharoufeh
%               e135a(Q,x,y,row,col) by Dr. Jeffrey Kharoufeh, Capt Cole Gulyas

tic                 % Start timer

t=10;
m=4;
n=2^m;
a=[1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
s=[1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0];
K= [3.8 .37 0              % Gamma
```

```
    2.1 .50  0              % Gamma
    2.2 .41 0               % Gamma
    3.2 .42  0              % Gamma
    1.2 .22  0              % Weibull
    1.8 .42  0              % Weibull
    1.9 .31  0              % Weibull
    1.2 .33 0               % Weibull
    1.8 2.1  2.3            % Triangular
    1.61 2.2  3.2           % Triangular
    1.1 1.83 2.4            % Triangular
    1.7 2.5  4.51           % Triangular
    .05 2.3  0              % Uniform
    .13 2.1  0              % Uniform
    .03 2.4  0              % Uniform
    .51  .93  0];           % Uniform

% Mean holding times for each state
mu=[]; for i=1:4
    mu=[mu; K(i,1)/K(i,2)];                               %Gamma
end for i=5:8
    mu=[mu; (1/K(i,2))*gamma((K(i,1)+1)/K(i,1))];       %Weibull
end for i=9:12
    mu=[mu; (K(i,1)+K(i,2)+K(i,3))/3];       %Triangular
end for i=13:16
    mu=[mu; (K(i,1)+K(i,2))/2];       %Uniform
end

% Rate matrix built with only function 1 repaired if function 2 is
% available
R=[-1/mu(1) .2/mu(1) .3/mu(1) .3/mu(1) .2/mu(1) 0 0 0 0 0 0 0 0 0 0 0
   .2/mu(2) -1/mu(2) 0 0 0 .4/mu(2) .2/mu(2) .2/mu(2) 0 0 0 0 0 0 0 0
   0 0 -1/mu(3) 0 0 .2/mu(3) 0 0 .4/mu(3) .4/mu(3) 0 0 0 0 0 0
   0 0 0 -1/mu(4) 0 0 .5/mu(4) 0 .3/mu(4) 0 .2/mu(4) 0 0 0 0 0
   0 0 0 0 -1/mu(5) 0 0 .2/mu(5) 0 .7/mu(5) .1/mu(5) 0 0 0 0 0
   0 0 .4/mu(6) 0 0 -1/mu(6) 0 0 0 0 0 .3/mu(6) .3/mu(6) 0 0 0
   0 0 0 .5/mu(7) 0 0 -1/mu(7) 0 0 0 0 .2/mu(7) 0 .3/mu(7) 0 0
   0 0 0 0 .4/mu(8) 0 0 -1/mu(8) 0 0 0 0 .5/mu(8) .1/mu(8) 0 0
   0 0 0 0 0 0 0 0 -1/mu(9) 0 0 .7/mu(9) 0 0 .3/mu(9) 0
   0 0 0 0 0 0 0 0 0 -1/mu(10) 0 0 .8/mu(10) 0 .2/mu(10) 0
   0 0 0 0 0 0 0 0 0 0 -1/mu(11) 0 0 .6/mu(11) .4/mu(11) 0
   0 0 0 0 0 0 0 0 .2/mu(12) 0 0 -1/mu(12) 0 0 0 .8/mu(12)
   0 0 0 0 0 0 0 0 0 .3/mu(13) 0 0 -1/mu(13) 0 0 .7/mu(13)
   0 0 0 0 0 0 0 0 0 0 .3/mu(14) 0 0 -1/mu(14) 0 .7/mu(14)
```

```
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1/mu(15) 1/mu(15)
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 eps];

% P*(s)
P_Laplace_SMP=prob_matrix(R,t);

% Pmf
pmf_SMP=a*P_Laplace_SMP;

% Availability
A_SMP=pmf_SMP*s';

duration=toc       % Stop timer

% Calculate stationary distribution and availability
% Pi
ze=zeros(1,n);
ze(end)=1;
Pe=[];
for i=1:n
    for j=1:n
        if i~=j Pe(i,j)=-R(i,j)/R(i,i); end
    end
end
Pe=Pe-eye(n);
Pe(:,end)=ones(n,1);
pie=ze*inv(Pe);

% Pmf and stationary availability
p=[]; for j=1:n
    p=[p (pie(j)*mu(j))/(pie*(mu))];
end A_stat=p*s';

%-----------Sub Function------------------------------------------------
function eq = e135a(Q,x,y,row,col)
% Author:           Capt Gulyas, Dr. Jeffrey Kharoufeh
% Last Revision:    4 Oct 04
% Note: function handles Q, row, and col were added to function e135a

s=x+y*i;                    % Complex transform variable
P_star = [];                % Declare matrix
I=eye(size(Q));             % Identity matrix
```

```
K= [3.8 .37 0               % Gamma
    2.1 .50  0              % Gamma
    2.2 .41 0               % Gamma
    3.2 .42  0              % Gamma
    1.2 .22  0              % Weibull
    1.8 .42  0              % Weibull
    1.9 .31  0              % Weibull
    1.2 .33 0               % Weibull
    1.8 2.1  2.3            % Triangular
    1.61 2.2  3.2           % Triangular
    1.1 1.83 2.4            % Triangular
    1.7 2.5  4.51           % Triangular
    .05 2.3  0              % Uniform
    .13 2.1  0              % Uniform
    .03 2.4  0              % Uniform
    .51  .93  0];           % Uniform

LSTG=[]; LSTD=[];

% States 1-4: Gamma
for a=1:4
    F_lst=(K(a,2)/(s+K(a,2)))^K(a,1);
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

% States 5-8: Weibull
for a=5:8
    sumterm=0;
    N=1000;
    for k=1:N
        sumterm=sumterm + exp(-s*(1/K(a,2))*(-log(1-(k/(N+1))))^(1/K(a,1)));
    end
    F_lst=(1/N)*sumterm;
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

% States 9-12: Triangular
```

```
for a=9:12
    term1=(exp(-s*K(a,1))-exp(-s*K(a,2)))/((K(a,3)-K(a,1))*(K(a,2)-K(a,1)));
    term2=(exp(-s*K(a,3))-exp(-s*K(a,2)))/((K(a,3)-K(a,1))*(K(a,3)-K(a,2)));
    F_lst=(2/(s^2))*(term1+term2);
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

% States 13-16: Uniform
for a=13:16
    F_lst=(exp(-s*K(a,1))-exp(-s*K(a,2)))/(s*(K(a,2)-K(a,1)));
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

P_star = (1/s)*(inv(I-LSTG))*LSTD;        % Build P*(s)
eq = real(P_star(row,col));               % Return first moment
%-----------End Sub Function-------------------------------------------


function [t,A_SMP,A_stat,duration] = Availability_Example3_1(t)
% Author:       Capt Cole Gulyas
% Date:         22 Jan 05
% Input:        t=time
% Output:       Measures for Example 3, satellite 1: t=time, A_SMP=instantaneous
%               availability, A_stat=stationary availability, and duration=processing time.
% Subfunctions: prob_matrix(Q,t) by Capt Cole Gulyas, Dr. Jeffrey Kharoufeh
%               invt_lap(Q,t,row,col) by Dr. Jeffrey Kharoufeh
%               e135a(Q,x,y,row,col) by Dr. Jeffrey Kharoufeh, Capt Cole Gulyas

tic                % Start timer

m=4;
n=2^m;
a=[1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
s=[1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
K= [5.8 .37 0              % Gamma
    4.1 6.80  0            % Gamma
    1.6 2.31  0            % Gamma
    1.2 1.13 0             % Gamma
```

```
    2 7.41 0                % Erlang
    5 6.42  0               % Erlang
    1.2 1.22  0             % Weibull
    1.8 2.2  0              % Weibull
    .05 1.4  0              % Uniform
    .13 1.1  0              % Uniform
    .02 .5  1.3             % Triangular
    .13 .4  1.2             % Triangular
    .09 .3  1.1             % Triangular
    1 1.6  0                % Exp
    1 1.4  0                % Exp
    1  1.8  0 ];            % Exp

% Mean holding times for each state
mu=[];
for i=1:6
    mu=[mu; K(i,1)/K(i,2)];                                 %Gamma & Erlang
end
for i=7:8
    mu=[mu; (1/K(i,2))*gamma((K(i,1)+1)/K(i,1))];           %Weibull
end
for i=9:10
    mu=[mu; (K(i,1)+K(i,2))/2];                             %Uniform
end
for i=11:13
    mu=[mu; (K(i,1)+K(i,2)+K(i,3))/3];                      %Triangular
end
for i=14:16
    mu=[mu; K(i,1)/K(i,2)];                                 %Exp
end

% Rate matrix built with all types of repairs possible
R=[-1/mu(1) .2/mu(1) .3/mu(1) .3/mu(1) .2/mu(1) 0 0 0 0 0 0 0 0 0 0 0
   .4/mu(2) -1/mu(2) 0 0 0 .2/mu(2) .2/mu(2) .2/mu(2) 0 0 0 0 0 0 0 0
   .5/mu(3) 0 -1/mu(3) 0 0 .1/mu(3) 0 0 .2/mu(3) .2/mu(3) 0 0 0 0 0 0
   .2/mu(4) 0 0 -1/mu(4) 0 0 .4/mu(4) 0 .2/mu(4) 0 .2/mu(4) 0 0 0 0 0
   .4/mu(5) 0 0 0 -1/mu(5) 0 0 .1/mu(5) 0 .4/mu(5) .1/mu(5) 0 0 0 0 0
   0 .3/mu(6) .3/mu(6) 0 0 -1/mu(6) 0 0 0 0 0 .1/mu(6) .3/mu(6) 0 0 0
   0 .2/mu(7) 0 .4/mu(7) 0 0 -1/mu(7) 0 0 0 0 .2/mu(7) 0 .2/mu(7) 0 0
   0 .3/mu(8) 0 0 .3/mu(8) 0 0 -1/mu(8) 0 0 0 0 .3/mu(8) .1/mu(8) 0 0
   0 0 .3/mu(9) .3/mu(9) 0 0 0 0 -1/mu(9) 0 0 .3/mu(9) 0 0 .1/mu(9) 0
   0 0 .2/mu(10) 0 .3/mu(10) 0 0 0 0 -1/mu(10) 0 0 .3/mu(10) 0 .2/mu(10) 0
   0 0 0 .3/mu(11) .1/mu(11) 0 0 0 0 0 -1/mu(11) 0 0 .3/mu(11) .3/mu(11) 0
```

```
   0 0 0 0 0 .2/mu(12) .3/mu(12) 0 .2/mu(12) 0 0 -1/mu(12) 0 0 0 .3/mu(12)
   0 0 0 0 0 .3/mu(13) 0 .3/mu(13) 0 .1/mu(13) 0 0 -1/mu(13) 0 0 .3/mu(13)
   0 0 0 0 0 0 .2/mu(14) .4/mu(14) 0 0 .3/mu(14) 0 0 -1/mu(14) 0 .1/mu(14)
   0 0 0 0 0 0 0 0 .2/mu(15) .3/mu(15) .3/mu(15) 0 0 0 -1/mu(15) .2/mu(15)
   0 0 0 0 0 0 0 0 0 0 0 .4/mu(16) .4/mu(16) .1/mu(16) .1/mu(16) -1/mu(16)];

% P*(s)
P_Laplace_SMP=prob_matrix(R,t);

% Pmf
pmf_SMP=a*P_Laplace_SMP;

% Availability
A_SMP=pmf_SMP*s';

duration=toc;       % Stop timer

% Calculate stationary distribution and availability
% Pi
ze=zeros(1,n);
ze(end)=1;
Pe=[];
for i=1:n
    for j=1:n
        if i~=j Pe(i,j)=-R(i,j)/R(i,i); end
    end
end
Pe=Pe-eye(n);
Pe(:,end)=ones(n,1);
pie=ze*inv(Pe);

% Pmf and stationary availability
p=[];
for j=1:n
    p=[p (pie(j)*mu(j))/(pie*(mu))];
end
A_stat=p*s';

%-----------Sub Function------------------------------------------------
function eq = e135a(Q,x,y,row,col)
% Author:           Capt Gulyas, Dr. Jeffrey Kharoufeh
% Last Revision:    4 Oct 04
% Note: function handles Q, row, and col were added to function e135a
```

```
s=x+y*i;                                % Complex transform variable
P_star = [];                            % Declare matrix
I=eye(size(Q));                         % Identity matrix

K= [5.8 .37 0               % Gamma
    4.1 6.80  0             % Gamma
    1.6 2.31  0             % Gamma
    1.2 1.13 0              % Gamma
    2 7.41 0                % Erlang
    5 6.42  0               % Erlang
    1.2 1.22  0             % Weibull
    1.8 2.2  0              % Weibull
    .05 1.4  0              % Uniform
    .13 1.1  0              % Uniform
    .02 .5  1.3             % Triangular
    .13 .4  1.2             % Triangular
    .09 .3  1.1             % Triangular
    1 1.6  0                % Exp
    1 1.4  0                % Exp
    1  1.8  0 ];            % Exp

LSTG=[];
LSTD=[];

% States 1-6: Gamma and Erlang
for a=1:6
    F_lst=(K(a,2)/(s+K(a,2)))^K(a,1);
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

% States 7-8: Weibull
for a=7:8
    sumterm=0;
    N=1000;
    for k=1:N
        sumterm=sumterm + exp(-s*(1/K(a,2))*(-log(1-(k/(N+1))))^(1/K(a,1)));
    end
    F_lst=(1/N)*sumterm;
    for b=1:length(Q)
```

```
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

% States 9-10: Uniform
for a=9:10
    F_lst=(exp(-s*K(a,1))-exp(-s*K(a,2)))/(s*(K(a,2)-K(a,1)));
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

% States 11-13: Triangular
for a=11:13
    term1=(exp(-s*K(a,1))-exp(-s*K(a,2)))/((K(a,3)-K(a,1))*(K(a,2)-K(a,1)));
    term2=(exp(-s*K(a,3))-exp(-s*K(a,2)))/((K(a,3)-K(a,1))*(K(a,3)-K(a,2)));
    F_lst=(2/(s^2))*(term1+term2);
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

% States 14-16: Exponential
for a=14:16
    F_lst=(K(a,2)/(s+K(a,2)))^K(a,1);
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

P_star = (1/s)*(inv(I-LSTG))*LSTD;        % Build P*(s)
eq = real(P_star(row,col));               % Return first moment
%-----------End Sub Function-------------------------------------------


function [t,A_SMP,A_stat,duration] = Availability_Example3_2(t)
% Author:       Capt Cole Gulyas
% Date:         22 Jan 05
% Input:        t=time
```

```
% Output:       Measures for Example 3, satellite 2: t=time, A_SMP=instantaneous
%               availability, A_stat=stationary availability, and duration=processing time.
% Subfunctions: prob_matrix(Q,t) by Capt Cole Gulyas, Dr. Jeffrey Kharoufeh
%               invt_lap(Q,t,row,col) by Dr. Jeffrey Kharoufeh
%               e135a(Q,x,y,row,col) by Dr. Jeffrey Kharoufeh, Capt Cole Gulyas

tic                 % Start timer

m=3;
n=2^m;
a=[1 0 0 0 0 0 0 0];
s=[1 0 0 0 0 0 0 0];
K= [8.7 1.07 0              % Gamma
    3.1 5.80  0             % Gamma
    2 6.41 0                % Erlang
    5 6.42  0               % Erlang
    1.2 1.22  0             % Weibull
    1.8 2.2  0              % Weibull
    .02 .5  1.3             % Triangular
    1  1.8  0 ];            % Exp

% Mean holding times for each state
mu=[];
for i=1:4
    mu=[mu; K(i,1)/K(i,2)];                                 %Gamma & Erlang
end
for i=5:6
    mu=[mu; (1/K(i,2))*gamma((K(i,1)+1)/K(i,1))];           %Weibull
end
for i=7
    mu=[mu; (K(i,1)+K(i,2)+K(i,3))/3];                      %Triangular
end
for i=8
    mu=[mu; K(i,1)/K(i,2)];                                 %Exp
end

% Rate matrix built with all types of repairs possible
% R matrix (observed rates of state transitions)
R=[ -1/mu(1) .5/mu(1) .3/mu(1) .2/mu(1) 0 0 0 0
    .4/mu(2) -1/mu(2) 0 0 .3/mu(2) .3/mu(2) 0 0
    .3/mu(3) 0 -1/mu(3) 0 .3/mu(3) 0 .4/mu(3) 0
    .3/mu(4) 0 0 -1/mu(4) 0 .5/mu(4) .2/mu(4) 0
    0 .5/mu(5) .4/mu(5) 0 -1/mu(5) 0 0 .1/mu(5)
```

```
    0 .4/mu(6) 0 .3/mu(6) 0 -1/mu(6) 0 .3/mu(6)
    0 0 .4/mu(7) .3/mu(7) 0 0 -1/mu(7) .3/mu(7)
    0 0 0 0 .3/mu(8) .3/mu(8) .4/mu(8) -1/mu(8)];

% P*(s)
P_Laplace_SMP=prob_matrix(R,t)

% Pmf
pmf_SMP=a*P_Laplace_SMP

% Availability
A_SMP=pmf_SMP*s'

duration=toc;       % Stop timer

% Calculate stationary distribution and availability
% Pi
ze=zeros(1,n);
ze(end)=1;
Pe=[];
for i=1:n
    for j=1:n
        if i~=j Pe(i,j)=-R(i,j)/R(i,i); end
    end
end
Pe=Pe-eye(n);
Pe(:,end)=ones(n,1);
pie=ze*inv(Pe);

% Pmf and stationary availability
p=[];
for j=1:n
    p=[p (pie(j)*mu(j))/(pie*(mu))];
end
A_stat=p*s';

%-----------Sub Function----------------------------------------------
function eq = e135a(Q,x,y,row,col)
% Author:           Capt Gulyas, Dr. Jeffrey Kharoufeh
% Last Revision:    4 Oct 04
% Note: function handles Q, row, and col were added to function e135a

s=x+y*i;                              % Complex transform variable
```

```
P_star = [];                            % Declare matrix
I=eye(size(Q));                         % Identity matrix

K= [8.7 1.07 0              % Gamma
    3.1 5.80  0             % Gamma
    2 6.41 0                % Erlang
    5 6.42  0               % Erlang
    1.2 1.22  0             % Weibull
    1.8 2.2  0              % Weibull
    .02 .5  1.3             % Triangular
    1  1.8  0 ];            % Exp

LSTG=[];
LSTD=[];

% States 1-4: Gamma and Erlang
for a=1:4
    F_lst=(K(a,2)/(s+K(a,2)))^K(a,1);
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

% States 5-6: Weibull
for a=5:6
    sumterm=0;
    N=1000;
    for k=1:N
        sumterm=sumterm + exp(-s*(1/K(a,2))*(-log(1-(k/(N+1))))^(1/K(a,1)));
    end
    F_lst=(1/N)*sumterm;
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

% State 7: Triangular
for a=7
    term1=(exp(-s*K(a,1))-exp(-s*K(a,2)))/((K(a,3)-K(a,1))*(K(a,2)-K(a,1)));
    term2=(exp(-s*K(a,3))-exp(-s*K(a,2)))/((K(a,3)-K(a,1))*(K(a,3)-K(a,2)));
    F_lst=(2/(s^2))*(term1+term2);
```

```
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

% State 8: Exponential
for a=8
    F_lst=(K(a,2)/(s+K(a,2)))^K(a,1);
    for b=1:length(Q)
        if a~=b LSTG(a,b)=(Q(a,b)/(-Q(a,a)))*F_lst; end
        LSTD(a,a)=1-F_lst;
    end
end

P_star = (1/s)*(inv(I-LSTG))*LSTD;        % Build P*(s)
eq = real(P_star(row,col));               % Return first moment
%----------End Sub Function------------------------------------------


function f1 = invt_lap(Q,t,row,col)
% The purpose of this MATLAB program is to approximate the inverse transform of a one-
% dimensional Laplace transform in order to find the moments of the probability
% distribution, G(t). The program is based on the algorithm of Abate and Whitt (1995).
%
%       Author:  Jeffrey P. Kharoufeh, Ph.D. Candidate, IE & OR, Penn State University
%         Date:  January 23, 2001
% Last Revised:  February 5, 2001
%   References:  Abate, J. and W. Whitt (1995).  Numerical Inversion of the Laplace
%                Transform of Probability Distribution. ORSA Journal on Computing, 7,
%                36-43.

% Note: function handles Q, row, and col were added to function invt_lap

%Initialize variables, set parameters
rho=0.8; qx=[0.8]; tx=[0]; m=11; c=[]; ga=8; A=ga*log(10); mm=2^m;

for k=0:m
   d=nchoosek(m,k);
   c=[c d];
end
for t = t;
   tx = t;
   ntr=15;
```

```
   u=exp(A/2)/t;
   x=A/(2*t);
   h=pi/t;
   su=zeros(m+2);
   sm=e135a(Q,x,0,row,col)/2;
   for k=1:ntr
      y=k*h;
      sm=sm+((-1)^k)*e135a(Q,x,y,row,col);
   end
   su(1)=sm;
   for k=1:12
      n=ntr+k;
      y=n*h;
      su(k+1)=su(k)+((-1)^n)*e135a(Q,x,y,row,col);
   end
   av1=0; av2=0;
   for k=1:12
      av1=av1+c(k)*su(k);
      av2=av2+c(k)*su(k+1);
   end
   f1 = u*av1/mm; f2=u*av2/mm; qx=[qx f2];
end


function P = prob_matrix(Q,t)
% Authors:      Capt Cole Gulyas, Dr. Jeffrey Kharoufeh
% Date:         27 Oct 04
% Input:        Q=rate transition matrix, t=Time, and eventually state
%               distributions, etc.
% Output:       Approximation of P(t), the probability transition matrix, using
%               functions INVT_LAP and e135 created by Dr. Kharoufeh as subfunctions.
% References:   Kharoufeh, Jeffrey.  INVT_LAP.  MatLab code.  Feb 5, 2001.
%               Kharoufeh, Jeffrey.  e135.m.  MatLab code.  Sept. 22, 2004.
%               See references for subfunctions.

format long

% Declare matrix
P=[];

%Build P matrix elements (required since INVT_LAP subfunction is univariate)
for i = 1:length(Q)
    for j=1:length(Q)
        disp(['(' num2str(i) ',' num2str(j) ')'])
```

```
        P(i,j)=invt_lap(Q,t,i,j);
    end
end


function Q = rate_matrix(m,f,r)
% Author:       Capt Cole Gulyas
% Date:         25 Sep 04
% Input:        m=number of functions in a satellite, f=1xm vector of
%               function failure rates, r=1xm vector of repair rates.
% Output:       Rate matrix Q.

% Initialize the storage array for S.
S=[];

% Determine dimension of S.
dimS=2^m;

% Create a matrix were the rows will be permuted to represent the
% combinations of states of failed functions
X=triu([ones(m,m);zeros(1,m)]);

% Create a matrix S where each row represents a possible combination of the
% state of the satellite.
for i=1:length(X)
    P=perms(X(i,:));            % Create permutations of having i-1 function down
    S=[S; unique(P,'rows')];    % Delete redundant combinations and append to S
end

% Build the transistion rate generator matrix.
Q=[];                           % Initialize Q

% Iterate through the elements of S finding all commutative states.  When a
% state is found, insert the corresponding rate into the Q matrix.
for i=1:dimS
    for j=1:dimS
        % Find an instance where the number of functions has increased or
        % decreased by 1.
        if sum(S(i,:)==S(j,:))==(m-1)
            y=S(i,:)-S(j,:);
            for k=1:m                   % Determine which function changed.
                if y(1,k)==1
                    Q(i,j)=f(k);        % Function failed.
                elseif y(1,k)==-1
```

```
                    Q(i,j)=r(k);        % Function was repaired.
                end
            end
        end
    end
end

% Write in diagonals on condition of row sums equal to zero.
for i=1:dimS
    Q(i,i)=-sum(Q(i,:));
end
```

A.2 Simulated Availability

```
function [t,A_sim, duration] = Sim_Example1(t,reps)
%  Orig Author: Jeffrey P. Kharoufeh, Ph.D. candidate, IE & OR, Penn State University
%         Date: January 15, 2001
%   Revised by: Captain Chris Solo, M.S. candidate, OR, Air Force Institute of Technology
%         Date: 29 January 2004
%   Revised by: Captain Cole Gulyas, M.S. candidate, OR, Air Force Institute of Technology
%         Date: 10 November 2004
%
% The purpose of this MATLAB program is to simulate a finite-state semi-Markov process.
% The process is simulated in order to validate the probability mass function at a specific
% time gained from transient analysis a satellite system modelled as an SMP.
% The program uses function "rando" in order to select the next state after a state transition.
%
% Input:        Q=Infinitesimal generator matrix, t=time associated with probability
%               mass function, a=initial state vector, and reps=number of
%               simulation repetitions
% Output:       Measures for Example 1: t=time, A_sim=simulated instantaneous availability,
%               and duration=processing time.

%1-4 Block A
m=3;
n=2^m;
a=[1 0 0 0 0 0 0 0];
s=[1 0 0 0 0 0 0 0];
f=[.0291 .0503 .0230];
r=[0 3.33 0];
```

```
% %5-16 Block A
% m=3;
% n=2^m;
% a=[1 0 0 0 0 0 0 0];
% s=[1 0 0 0 0 0 0 0];
% f=[.0121 .0421 .0189];
% r=[0 4.8216 0];

% %17-20 Block R
% m=3;
% n=2^m;
% a=[.98 0 .02 0 0 0 0 0];
% s=[1 0 0 0 0 0 0 0];
% f=[.0121 .0303 .0143];
% r=[0 4.9216 0];

% %21-24 Block R
% m=3;
% n=2^m;
% a=[1 0 0 0 0 0 0 0];
% s=[1 0 0 0 0 0 0 0];
% f=[.0103 .0282 .0118];
% r=[0 4.9216 0];

Q=rate_matrix(m,f,r);

% Convert mean holding times for each state from lambda form to mu
K=-1./diag(Q);

tic                 % Start timer

% Probability transition matrix
P=zeros(n,n);
for l=1:n
    for m=1:n
        if l~=m
            P(l,m)=(Q(l,m)/(-Q(l,l)));
        end
    end
end

statecounter = zeros(1,n);       % Counts number of times process was found in each state
```

```
for k = 1:reps
    if (mod(k,10000)==0) disp(['k = ' num2str(k)]); end      % Display number of reps
    Z = [];
    Z(1) = rando(a);                 % Initial state of the environment at time 0
    newtime = 0;

    % Specify the distribution for the initial state, corresponding to vector a
    totaltime = exprnd(K(Z(1)));        % ****** Time spent in initial state
    i=1;

    while (totaltime < t)
        Z(i+1) = rando(P(Z(i),:));      % Use P matrix to determine next state
        switch Z(i+1)
            case {1}
                newtime = exprnd(K(Z(i+1)));
            case {2}
                newtime = exprnd(K(Z(i+1)));
            case {3}
                newtime = exprnd(K(Z(i+1)));
            case {4}
                newtime = exprnd(K(Z(i+1)));
            case {5}
                newtime = exprnd(K(Z(i+1)));
            case {6}
                newtime = exprnd(K(Z(i+1)));
            case {7}
                newtime = exprnd(K(Z(i+1)));
            case {8}
                newtime = exprnd(K(Z(i+1)));
        end
        totaltime = totaltime + newtime;
        i=i+1;
    end
    statecounter(Z(end)) = statecounter(Z(end)) + 1;
end

pmf= statecounter./reps;
A_sim=pmf*s';
duration=toc


function [t,A_sim, duration] = Sim_Example2(t,reps)
%  Orig Author: Jeffrey P. Kharoufeh, Ph.D. candidate, IE & OR, Penn State University
%         Date: January 15, 2001
```

```
%   Revised by: Captain Chris Solo, M.S. candidate, OR, Air Force Institute of Technology
%         Date: 29 January 2004
%   Revised by: Captain Cole Gulyas, M.S. candidate, OR, Air Force Institute of Technology
%         Date: 6 January 2005
%
% The purpose of this MATLAB program is to simulate a finite-state semi-Markov process.
% The process is simulated in order to demonstrate the probability mass function at a specific
% time gained from transient analysis of a satellite system modelled as an SMP.
% The program uses function "rando" in order to select the next state after a state transition.
%
% Input:        t=time associated with probability mass function
%               reps=number of simulation repetitions
% Output:       Measures for Example 2: t=time, A_sim=simulated instantaneous availability,
%               and duration=processing time.

m=4;
n=2^m;
a=[1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
s=[1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0];
K= [3.8 .37 0               % Gamma
    2.1 .50  0              % Gamma
    2.2 .41 0               % Gamma
    3.2 .42  0              % Gamma
    1.2 .22  0              % Weibull
    1.8 .42  0              % Weibull
    1.9 .31  0              % Weibull
    1.2 .33 0               % Weibull
    1.8 2.1  2.3            % Triangular
    1.61 2.2  3.2           % Triangular
    1.1 1.83 2.4            % Triangular
    1.7 2.5  4.51           % Triangular
    .05 2.3  0              % Uniform
    .13 2.1  0              % Uniform
    .03 2.4  0              % Uniform
    .51  .93  0];           % Uniform

% Mean holding times for each state
mu=[];
for i=1:n
    mu=[mu; (1/K(i,2))*gamma((K(i,1)+1)/K(i,1))];       %k/lamda
end

R=[-1/mu(1) .2/mu(1) .3/mu(1) .3/mu(1) .2/mu(1) 0 0 0 0 0 0 0 0 0 0 0
```

```
   .2/mu(2) -1/mu(2) 0 0 0 .4/mu(2) .2/mu(2) .2/mu(2) 0 0 0 0 0 0 0 0
   0 0 -1/mu(3) 0 0 .2/mu(3) 0 0 .4/mu(3) .4/mu(3) 0 0 0 0 0 0
   0 0 0 -1/mu(4) 0 0 .5/mu(4) 0 .3/mu(4) 0 .2/mu(4) 0 0 0 0 0
   0 0 0 0 -1/mu(5) 0 0 .2/mu(5) 0 .7/mu(5) .1/mu(5) 0 0 0 0 0
   0 0 .4/mu(6) 0 0 -1/mu(6) 0 0 0 0 0 .3/mu(6) .3/mu(6) 0 0 0
   0 0 0 .5/mu(7) 0 0 -1/mu(7) 0 0 0 0 .2/mu(7) 0 .3/mu(7) 0 0
   0 0 0 0 .4/mu(8) 0 0 -1/mu(8) 0 0 0 0 .5/mu(8) .1/mu(8) 0 0
   0 0 0 0 0 0 0 0 -1/mu(9) 0 0 .7/mu(9) 0 0 .3/mu(9) 0
   0 0 0 0 0 0 0 0 0 -1/mu(10) 0 0 .8/mu(10) 0 .2/mu(10) 0
   0 0 0 0 0 0 0 0 0 0 -1/mu(11) 0 0 .6/mu(11) .4/mu(11) 0
   0 0 0 0 0 0 0 0 .2/mu(12) 0 0 -1/mu(12) 0 0 0 .8/mu(12)
   0 0 0 0 0 0 0 0 0 .3/mu(13) 0 0 -1/mu(13) 0 0 .7/mu(13)
   0 0 0 0 0 0 0 0 0 0 .3/mu(14) 0 0 -1/mu(14) 0 .7/mu(14)
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 -1/mu(15) 1/mu(15)
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 eps];

% Convert Gamma parameters from f(x|k,lambda) to f(x|k,beta)
% in order to use the matlab function weibrnd
for i=1:4
    K(i,:)=[K(i,1)  1/K(i,2) 0];
end

% Convert Weibull parameters from f(x|alpha,lambda) to f(x|a,b)
% in order to use the matlab function weibrnd
for i=5:8
    K(i,:)=[K(i,2)^K(i,1)  K(i,1) 0];
end

tic                 % Start timer

% Probability transition matrix
P=zeros(n,n);
for l=1:n
    for m=1:n
        if l~=m
            P(l,m)=(R(l,m)/(-R(l,l)));
        end
    end
end

statecounter = zeros(1,n);       % Counts number of times process was found in each state

for k = 1:reps
```

```
if (mod(k,1000)==0) disp(['k = ' num2str(k)]); end      % Display number of reps
Z = [];
Z(1) = rando(a);                % Initial state of the environment at time 0
newtime = 0;

% Specify the distribution for the initial state, corresponding to vector a
totaltime = gamrnd(K(Z(1),1),K(Z(1),2));       % ****** Time spent in initial state
i=1;

while (totaltime < t)
    Z(i+1) = rando(P(Z(i),:));      % Use P matrix to determine next state
    switch Z(i+1)
        case {1}
            newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
        case {2}
            newtime = gamrnd(K(Z(i+1),1),1/K(Z(i+1),2));
        case {3}
            newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
        case {4}
            newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
        case {5}
            newtime = weibrnd(K(Z(i+1),1),K(Z(i+1),2));
        case {6}
            newtime = weibrnd(K(Z(i+1),1),K(Z(i+1),2));
        case {7}
            newtime = weibrnd(K(Z(i+1),1),K(Z(i+1),2));
        case {8}
            newtime = weibrnd(K(Z(i+1),1),K(Z(i+1),2));
        case {9}
            newtime = trirnd(K(Z(i+1),1),K(Z(i+1),2),K(Z(i+1),3));
        case {10}
            newtime = trirnd(K(Z(i+1),1),K(Z(i+1),2),K(Z(i+1),3));
        case {11}
            newtime = trirnd(K(Z(i+1),1),K(Z(i+1),2),K(Z(i+1),3));
        case {12}
            newtime = trirnd(K(Z(i+1),1),K(Z(i+1),2),K(Z(i+1),3));
        case {13}
            newtime = unifrnd(K(Z(i+1),1),K(Z(i+1),2));
        case {14}
            newtime = unifrnd(K(Z(i+1),1),K(Z(i+1),2));
        case {15}
            newtime = unifrnd(K(Z(i+1),1),K(Z(i+1),2));
        case {16}
```

```
                newtime = unifrnd(K(Z(i+1),1),K(Z(i+1),2));
        end
        totaltime = totaltime + newtime;
        i=i+1;
    end
    statecounter(Z(end)) = statecounter(Z(end)) + 1;
end

pmf= statecounter./reps;
A_sim=pmf*s';
duration=toc


function [t,A_sim, duration] = Sim_Example3_1(t,reps)
%  Orig Author: Jeffrey P. Kharoufeh, Ph.D. candidate, IE & OR, Penn State University
%         Date: January 15, 2001
%   Revised by: Captain Chris Solo, M.S. candidate, OR, Air Force Institute of Technology
%         Date: 29 January 2004
%   Revised by: Captain Cole Gulyas, M.S. candidate, OR, Air Force Institute of Technology
%         Date: 6 January 2005
%
% The purpose of this MATLAB program is to simulate a finite-state semi-Markov process.
% The process is simulated in order to demonstrate the probability mass function at a specific
% time gained from transient analysis of a satellite system modelled as an SMP.
% The program uses function "rando" in order to select the next state after a state transition.
%
% Input:        t=time associated with probability mass function
%               reps=number of simulation repetitions
% Output:       Measures for Example 3, satellite 1: t=time, A_sim=simulated instantaneous
%               availability, and duration=processing time.

m=4;
n=2^m;
a=[1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
s=[1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
K= [5.8 .37 0               % Gamma
    4.1 6.80  0             % Gamma
    1.6 2.31  0             % Gamma
    1.2 1.13 0              % Gamma
    2 7.41 0                % Erlang
    5 6.42  0               % Erlang
    1.2 1.22  0             % Weibull
    1.8 2.2  0              % Weibull
    .05 1.4  0              % Uniform
```

```
    .13 1.1  0              % Uniform
    .02 .5  1.3             % Triangular
    .13 .4  1.2             % Triangular
    .09 .3  1.1             % Triangular
    1 1.6  0                % Exp
    1 1.4  0                % Exp
    1  1.8  0 ];            % Exp

% Mean holding times for each state
mu=[];
for i=1:6
    mu=[mu; K(i,1)/K(i,2)];                                 %Gamma & Erlang
end
for i=7:8
    mu=[mu; (1/K(i,2))*gamma((K(i,1)+1)/K(i,1))];           %Weibull
end
for i=9:10
    mu=[mu; (K(i,1)+K(i,2))/2];                             %Uniform
end
for i=11:13
    mu=[mu; (K(i,1)+K(i,2)+K(i,3))/3];                      %Triangular
end
for i=14:16
    mu=[mu; K(i,1)/K(i,2)];                                 %Exp
end

% Rate matrix built with all types of repairs possible
R=[-1/mu(1) .2/mu(1) .3/mu(1) .3/mu(1) .2/mu(1) 0 0 0 0 0 0 0 0 0 0 0
   .4/mu(2) -1/mu(2) 0 0 0 .2/mu(2) .2/mu(2) .2/mu(2) 0 0 0 0 0 0 0 0
   .5/mu(3) 0 -1/mu(3) 0 0 .1/mu(3) 0 0 .2/mu(3) .2/mu(3) 0 0 0 0 0 0
   .2/mu(4) 0 0 -1/mu(4) 0 0 .4/mu(4) 0 .2/mu(4) 0 .2/mu(4) 0 0 0 0 0
   .4/mu(5) 0 0 0 -1/mu(5) 0 0 .1/mu(5) 0 .4/mu(5) .1/mu(5) 0 0 0 0 0
   0 .3/mu(6) .3/mu(6) 0 0 -1/mu(6) 0 0 0 0 0 .1/mu(6) .3/mu(6) 0 0 0
   0 .2/mu(7) 0 .4/mu(7) 0 0 -1/mu(7) 0 0 0 0 .2/mu(7) 0 .2/mu(7) 0 0
   0 .3/mu(8) 0 0 .3/mu(8) 0 0 -1/mu(8) 0 0 0 0 .3/mu(8) .1/mu(8) 0 0
   0 0 .3/mu(9) .3/mu(9) 0 0 0 0 -1/mu(9) 0 0 .3/mu(9) 0 0 .1/mu(9) 0
   0 0 .2/mu(10) 0 .3/mu(10) 0 0 0 0 -1/mu(10) 0 0 .3/mu(10) 0 .2/mu(10) 0
   0 0 0 .3/mu(11) .1/mu(11) 0 0 0 0 0 -1/mu(11) 0 0 .3/mu(11) .3/mu(11) 0
   0 0 0 0 0 .2/mu(12) .3/mu(12) 0 .2/mu(12) 0 0 -1/mu(12) 0 0 0 .3/mu(12)
   0 0 0 0 0 .3/mu(13) 0 .3/mu(13) 0 .1/mu(13) 0 0 -1/mu(13) 0 0 .3/mu(13)
   0 0 0 0 0 0 .2/mu(14) .4/mu(14) 0 0 .3/mu(14) 0 0 -1/mu(14) 0 .1/mu(14)
   0 0 0 0 0 0 0 0 .2/mu(15) .3/mu(15) .3/mu(15) 0 0 0 -1/mu(15) .2/mu(15)
   0 0 0 0 0 0 0 0 0 0 0 .4/mu(16) .4/mu(16) .1/mu(16) .1/mu(16) -1/mu(16)];
```

```
% Convert Gamma parameters from f(x|k,lambda) to f(x|k,beta)
% in order to use the matlab function gamrnd
for i=1:6
    K(i,:)=[K(i,1)  1/K(i,2) 0];
end

% Convert Weibull parameters from f(x|alpha,lambda) to f(x|a,b)
% in order to use the matlab function weibrnd
for i=7:8
    K(i,:)=[K(i,2)^K(i,1)  K(i,1) 0];
end

% Convert Exp parameters from f(x|lambda) to f(x|beta)
% in order to use the matlab function gamrnd
for i=14:16
    K(i,:)=[K(i,1)  1/K(i,2) 0];
end

tic                 % Start timer

% Probability transition matrix
P=zeros(n,n);
for l=1:n
    for m=1:n
        if l~=m
            P(l,m)=(R(l,m)/(-R(l,l)));
        end
    end
end

statecounter = zeros(1,n);       % Counts number of times process was found in each state

for k = 1:reps
    if (mod(k,1000)==0) disp(['k = ' num2str(k)]); end      % Display number of reps
    Z = [];
    Z(1) = rando(a);                 % Initial state of the environment at time 0
    newtime = 0;

    % Specify the distribution for the initial state, corresponding to vector a
    totaltime = gamrnd(K(Z(1),1),K(Z(1),2));        % ****** Time spent in initial state
    i=1;
```

```
    while (totaltime < t)
        Z(i+1) = rando(P(Z(i),:));      % Use P matrix to determine next state
        switch Z(i+1)
            case {1}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {2}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {3}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {4}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {5}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {6}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {7}
                newtime = weibrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {8}
                newtime = weibrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {9}
                newtime = unifrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {10}
                newtime = unifrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {11}
                newtime = trirnd(K(Z(i+1),1),K(Z(i+1),2),K(Z(i+1),3));
            case {12}
                newtime = trirnd(K(Z(i+1),1),K(Z(i+1),2),K(Z(i+1),3));
            case {13}
                newtime = trirnd(K(Z(i+1),1),K(Z(i+1),2),K(Z(i+1),3));
            case {14}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {15}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {16}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
        end
        totaltime = totaltime + newtime;
        i=i+1;
    end
    statecounter(Z(end)) = statecounter(Z(end)) + 1;
end

pmf= statecounter./reps;
```

```
A_sim=pmf*s';
duration=toc


function [t,A_sim, duration] = Sim_Example3_2(t,reps)
%  Orig Author: Jeffrey P. Kharoufeh, Ph.D. candidate, IE & OR, Penn State University
%         Date: January 15, 2001
%   Revised by: Captain Chris Solo, M.S. candidate, OR, Air Force Institute of Technology
%         Date: 29 January 2004
%   Revised by: Captain Cole Gulyas, M.S. candidate, OR, Air Force Institute of Technology
%         Date: 6 January 2005
%
% The purpose of this MATLAB program is to simulate a finite-state semi-Markov process.
% The process is simulated in order to demonstrate the probability mass function at a specific
% time gained from transient analysis of a satellite system modelled as an SMP.
% The program uses function "rando" in order to select the next state after a state transition.
%
% Input:        t=time associated with probability mass function
%               reps=number of simulation repetitions
% Output:       Measures for Example 3, satellite 2: t=time, A_sim=simulated instantaneous
%               availability, and duration=processing time.

m=3;
n=2^m;
a=[1 0 0 0 0 0 0 0];
s=[1 0 0 0 0 0 0 0];
K= [8.7 1.07 0            % Gamma
    3.1 5.80  0           % Gamma
    2 6.41 0              % Erlang
    5 6.42  0             % Erlang
    1.2 1.22  0           % Weibull
    1.8 2.2  0            % Weibull
    .02 .5  1.3           % Triangular
    1  1.8  0 ];          % Exp

% Mean holding times for each state
mu=[];
for i=1:4
    mu=[mu; K(i,1)/K(i,2)];                              %Gamma & Erlang
end
for i=5:6
    mu=[mu; (1/K(i,2))*gamma((K(i,1)+1)/K(i,1))];        %Weibull
end
for i=7
```

```
    mu=[mu; (K(i,1)+K(i,2)+K(i,3))/3];                      %Triangular
end
for i=8
    mu=[mu; K(i,1)/K(i,2)];                                 %Exp
end

% Rate matrix built with all types of repairs possible
% R matrix (observed rates of state transitions)
R=[ -1/mu(1) .5/mu(1) .3/mu(1) .2/mu(1) 0 0 0 0
    .4/mu(2) -1/mu(2) 0 0 .3/mu(2) .3/mu(2) 0 0
    .3/mu(3) 0 -1/mu(3) 0 .3/mu(3) 0 .4/mu(3) 0
    .3/mu(4) 0 0 -1/mu(4) 0 .5/mu(4) .2/mu(4) 0
    0 .5/mu(5) .4/mu(5) 0 -1/mu(5) 0 0 .1/mu(5)
    0 .4/mu(6) 0 .3/mu(6) 0 -1/mu(6) 0 .3/mu(6)
    0 0 .4/mu(7) .3/mu(7) 0 0 -1/mu(7) .3/mu(7)
    0 0 0 0 .3/mu(8) .3/mu(8) .4/mu(8) -1/mu(8)];

% Convert Gamma parameters from f(x|k,lambda) to f(x|k,beta)
% in order to use the matlab function weibrnd
for i=1:4
    K(i,:)=[K(i,1)  1/K(i,2) 0];
end

% Convert Weibull parameters from f(x|alpha,lambda) to f(x|a,b)
% in order to use the matlab function weibrnd
for i=5:6
    K(i,:)=[K(i,2)^K(i,1)  K(i,1) 0];
end

% Convert Gamma parameters from f(x|k,lambda) to f(x|k,beta)
% in order to use the matlab function weibrnd
for i=8
    K(i,:)=[K(i,1)  1/K(i,2) 0];
end

tic                 % Start timer

% Probability transition matrix
P=zeros(n,n);
for l=1:n
    for m=1:n
        if l~=m
            P(l,m)=(R(l,m)/(-R(l,l)));
```

```
        end
    end
end

statecounter = zeros(1,n);       % Counts number of times process was found in each state

for k = 1:reps
    if (mod(k,1000)==0) disp(['k = ' num2str(k)]); end      % Display number of reps
    Z = [];
    Z(1) = rando(a);                 % Initial state of the environment at time 0
    newtime = 0;

    % Specify the distribution for the initial state, corresponding to vector a
    totaltime = gamrnd(K(Z(1),1),K(Z(1),2));        % ****** Time spent in initial state
    i=1;

    while (totaltime < t)
        Z(i+1) = rando(P(Z(i),:));      % Use P matrix to determine next state
        switch Z(i+1)
            case {1}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {2}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {3}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {4}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {5}
                newtime = weibrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {6}
                newtime = weibrnd(K(Z(i+1),1),K(Z(i+1),2));
            case {7}
                newtime = trirnd(K(Z(i+1),1),K(Z(i+1),2),K(Z(i+1),3));
            case {8}
                newtime = gamrnd(K(Z(i+1),1),K(Z(i+1),2));
        end
        totaltime = totaltime + newtime;
        i=i+1;
    end
    statecounter(Z(end)) = statecounter(Z(end)) + 1;
end

pmf= statecounter./reps;
```

```
A_sim=pmf*s';
duration=toc



function [index] = rando(p)
% rando.m generates a discrete random variable in S={1,2,...,n} given
% a distribution vector p = [p1 p2 ... pn].

u = rand;
i = 1;
s = p(1);

while ((u > s) & (i < length(p))),
  i=i+1;
  s=s+p(i);
end
index=i;



function n = trirnd(a,m,b)
% Author:   Capt Cole Gulyas
% Date:     15 Nov 04
% Input:    a=min, m=mode, b=max
% Output:   Random number from triangular(a,m,b)

format long

d=(m-a)/(b-a);
r=rand(1);
if r<d
    n=sqrt(r*(b-a)*(m-a)) + a;
else
    n=b - sqrt((1-r)*(b-a)*(b-m));
end
```

Bibliography

1. Abate, J. and W. Whitt (1995). Numerical inversion of Laplace transforms of probability distributions. *ORSA Journal on Computing*, **7**, 36-43.

2. Barlow, R. (1984). Mathematical theory of reliability: A historical perspective. *IEEE Transactions of Reliability*. **1**, 16-20.

3. Barzilai, J. (2003). Advances in preference function modelling. *IEEE International Conference on Systems, Man and Cybernetics, 2003*. **5**, 4483-4489.

4. Campbell, N. (1941). The replacement of perishable members of a continually operating system. *Journal of the Royal Statistical Society*, **7**, 110-130.

5. Chairman of the Joint Chiefs of Staff. *Joint Doctrine for Space Operations*. Joint Publication 3-14. Washington: GPO, 9 August 2002.

6. Chairman of the Joint Chiefs of Staff. *Joint Vision 2020*. Washington: GPO, June 2000.

7. Clemens, R. and T. Reilly (2001). *Making Hard Decisions with DecisionTools*. Duxbury Press, California.

8. Collopy, P. (2003). Assigning value to reliability in satellite constellations. *AIAA Paper 2003-6214*.

9. Comstock, J. (2004). Deputy Program Director, Welkin Associates, Ltd., Chantilly, VA. Telephone interview. 28 June 2004.

10. Cosqueric, L., LeBlanc,P., Salvaterra, G., and L. Toudret (2000). Establishment of derating rules and end of life drifts figures for electronic components. *Microelectronics Reliability*, **40**, 1273-1278.

11. Daniel, C. and F. Wood (1980). Fitting Equations to Data: Computer Analysis of Multifactor Data. John Wiley & Sons, Inc., New York.

12. Davis, D. (1952). An analysis of some failure data. *Journal of the American Statistical Association*, **47**, 113-150.

13. Dubner, H. and J. Abate (1968). Numerical inversion of Laplace transforms by relating them to the finite Fourier-cosine transform. *Journal of the Association for Computing Machinery*, **15**, 115-123.

14. Ebeling, C. (1997). *An Introduction to Reliability and Maintainability Engineering*. McGraw-Hill Companies, Inc., New York.

15. Ebstein, B. and M. Sobel (1953). Life testing. *Journal of the American Statistical Association*, **48**, 486-502.

16. Edwards, C. and D. Penney (1989). *Elementary Differential Equations with Applications*. 3rd ed., Prentice-Hall, New Jersey.

17. Feuchter, C., C. Van Meter, K. Neuman, and K. Sparrow (1991). When is a satellite not a toaster? *Proceedings of the 1991 Winter Simulation Conference*, 499-508.

18. Fischer, M., D. Gross, D. Masi, J. Shortle (2001). Analyzing the waiting time process in internet queueing systems with the transform approximation method. *The Telecommunications Review*, 21-32.

19. Fricks, R. and K. Trivedi (1998). Availability modeling of energy management systems. *Microelectronics Reliability*, **38**, 727-743.

20. Fussell, J. and W. Vesely (1972). A new methodology for obtaining cut sets for falut trees. *Transactions of the American Nuclear Society*, **15**, 262-263

21. Goldstein, T., S. Ladany, and A. Mehrez (1988). A discounted machine-replacement model with an expected future technological breakthrough. *Naval Research Logistics Quarterly*, **35**, 209-220.

22. Harris, C., P. Brill, and M. Fischer (2000). Internet-type queues with power-tailed interarrival times and computational methods for their analysis. *INFORMS Journal on Computing*, **12**, 261-271.

23. Heyman, D. and M. Sobel (1982). *Stochastic Models in Operations Research, Volume I, Stochastic Processes and Operating Characteristics*. McGraw-Hill Book Company, New York.

24. Hofmann-Wellenhof, B., H. Lichtenegger, and J. Collins (2001). *GPS Theory and Practice*. 5th ed., Springer-Verlag Wien, New York.

25. Hopp, W. and S. Nair (1994). Markovian deterioration and technological change. *IIE Transactions*, **6**, 74-82.

26. Ibe, O., R. Howe, and K. Trivedi (1989). Approximate availability analysis of VAXcluster systems. *IEEE Transactions on Reliability*, **38**, 146-152.

27. Kao, J. (1956). A new life-quality measure for electron tubes. *IRE Transactions on Reliability and Quality Control*, **PGRQC-7**.

28. Kapur, K. and L. Lamberson (1977). *Reliability in Engineering Design*. John Wiley & Sons, Inc., New York.

29. Keeney, R. and H. Raiffa (1976). *Decisions with Multiple Objectives: Preferences and Value Tradeoffs*. John Wiley & Sons, Inc., New York.

30. Kulkarni, V. (1995). *Modeling and Analysis of Stochastic Systems*. Chapman & Hall, London.

31. Lotka, A. (1939). A contribution to the theory of self-renewing aggregates with special reference to industrial replacement. *The Annals of Mathematical Statistics*, **10**, 1-25.

32. Meyer, J. (1980). On evaluating the performability of degradable computing systems. *IEEE Transactions on Computers*, **C-27**, 720-731.

33. Shortle, J., M. Fischer, D. Gross, and D. Masi (2003). Using the transform approximation method to analyze queues with heavy-tailed service. *Journal of Probability and Statistical Science*, **1**, 15-27.

34. Smith, R., K. Trivedi, and A. Ramesh (1988). Performability analysis: measures, an algorithm, and a case study. *IEEE Transactions on Computers*, **37**, 406-417.

35. Staats, R. (1994). *A Multi-Attribute-Utility-Theory Model that Minimizes Interview-Data Requirements: A Consolidation of Space Launch Decisions.* M.S. Thesis. Department of Operational Sciences, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio.

36. "U.S. Air Force Fact Sheet, Defense Meteorological Satellite Program." Excerpt from unpublished article. n. pag. http://www.af.mil/factsheets/factsheet.asp?fsID=94. 22 July 2004.

37. "U.S. Air Force Fact Sheet, Milstar Satellite Communication System." Excerpt from unpublished article. n. pag. http://www.af.mil/factsheets/factsheet.asp?fsID=118. 18 January 2005.

38. "U.S. Air Force Fact Sheet, Navstar Global Positioning System." Excerpt from unpublished article. n. pag. http://www.af.mil/factsheets/factsheet.asp?fsID=119. 18 January 2005.

39. Wackerly, D., W. Mendenhall III, and R. Scheaffer (2002). *Mathematical Statistics with Applications.* 6th ed., Duxbury Press, California.

40. Weibull, W. (1951). A statistical distribution function of wide applicability. *Journal of Applied Mechanics*, **18**, 293-297.

41. Weiss, G. (1956). On the theory of replacement of machinery with a random failure time. *Naval Research Logistics Quarterly*, **3**, 279-293.

42. Wilson, S. (1990). *Measuring the Effectiveness of Space: Satellite Weather Systems.* M.S. Thesis. Department of Operational Sciences, Air Force Institute of Technology, Wright-Patterson Air Force Base, Ohio.

43. Zelen, M. and C. Dannemiller (1961). The robustness of life testing procedures derived from the exponential distribution. *Technometrics*, **3**, 29-49.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From – To)*
01-03-2005	**Master's Thesis**	Jun 2004 – Mar 2005

4. TITLE AND SUBTITLE

Stochastic Capability Models for Degrading Satellite Constellations

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Gulyas, Cole, W., Captain, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S)

Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Street, Building 642
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT/GOR/ENS/05-07

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

National Reconnaissance Office
Attn: William J. Comstock, Rm 43D19H
14675 Leed Road
Chantilly, VA 20151-1715 (703) 808-4436

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This thesis proposes and analyzes a new measure of functional capability for satellite constellations that incorporates the instantaneous availability and mission effectiveness of individual satellites. The capability measure yields a continuous score between zero and one accounting for the degree to which the constellation meets operational requirements. The measure is computed from an average of satellite capabilities, composed of the product of the satellite's instantaneous availability and value score. Instantaneous availability is acquired by modeling the satellite degradation status as either a time-homogenous, continuous-time Markov chain (CTMC) if it possesses functions with exponential lifetime distributions, or as a time-homogenous, semi-Markov process (SMP) if the function lifetime distributions are not exponential. The satellite value score represents the individual satellite's contribution to the overall constellation mission and is obtained using multi-attribute value theory. For illustrative purposes, analytical results were compared with those obtained via the Monte Carlo method and were found to be indistinguishable with substantially less computational effort.

15. SUBJECT TERMS

Markov process, semi-Markov process, reliability, availability, stochastic model, multi-attribute value theory, satellite constellations

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Jeffrey P. Kharoufeh, (ENS)
					19b. TELEPHONE NUMBER *(Include area code)*
U	U	U	UU	131	(937)255-6565, ext 4603; e-mail: Jeffrey.Kharoufeh@afit.edu